# GEMINI-10X/15X
# USERS MANUAL

# Your Gemini

# User's Manual

Not Intended
for Sale

**Trademark Acknowledgement**

**Gemini-10X, Gemini-15X, grafstar, Universal/Atari Parallel Interface, Universal/Commodore Parallel Interface:** Star Micronics, Inc.
**Apple, Apple II, Apple II+, Apple IIe, Apple III, Applesoft:** Apple Computer Inc.
**Atari 400, Atari 800, Atari 850:** Atari Inc., a Warner Communications Company
**Commodore, VIC-20, C-64:** Commodore Business Machines, Inc.
**CP/M:** Digital Research
**IBM Personal Computer, IBM PC:** International Business Machines Corporation
**Microsoft BASIC:** Microsoft Corporation
**Osborne 1:** Osborne Computer Corporation
**TRS-80:** Radio Shack, a division of Tandy Corporation

© 1983, Star Micronics, Inc.

# Preface

Congratulations on having purchased a serial, impact dot matrix printer that offers you more quality, greater flexibility and convenience, and superior reliability than you could find in any other printer in the same price range.

As you already know (or are about to learn), the Gemini-10X and Gemini-15X let you print both regular and italicized alphabets, as well as normal, elite (smaller), condensed, enlarged, emphasized, and double-strike characters. You can print on regular stationery, multiple-sheet forms, fan-fold computer paper, or paper fed from a roll (European style).

The graphics capability provided by the Gemini printers is limited only by your graphic talents, and includes three bit-image modes: Low Resolution (60 by 72 dots per square inch), High Resolution (120 by 144) and Ultra High Resolution (240 by 144). With all this printing power at your fingertips, what are you waiting for? Let's get started!

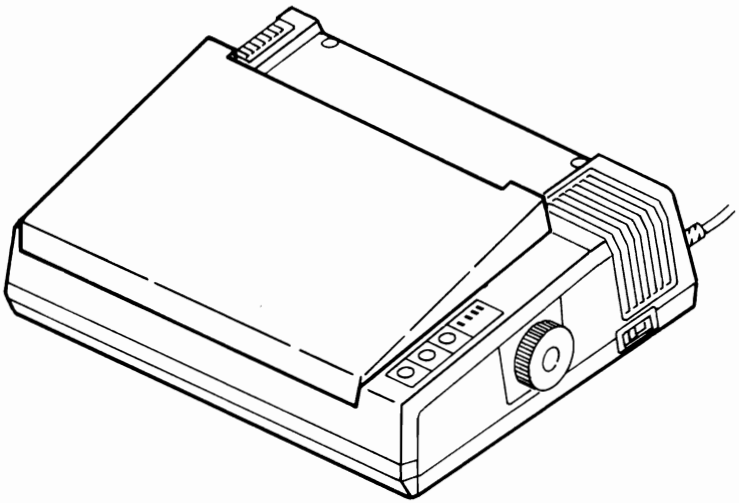
This User's Manual is organized to serve both the first-time user and the sophisticated computer buff. The first five chapters provide the basics you need for putting your printer to work in most home and office applications. Chapters 6 through 10 deliver information for intermediate and advanced users, supported by technically rich appendices. If you are a beginner, you might want to look at Chapters 6-10 sometime in the future, when your curiosity about how the printer does what it does has replaced your initial interest in what it can do.

Take a moment to examine the Table of Contents on the following page. It shows you how this manual is organized to best serve your information needs. You should note, if you are just now unpacking your new printer from its shipment packaging, that Appendix G shows you what you have and how it all fits together. This manual's first several appendices are computer-specific, and help you connect your new Gemini with your computer. At the back of this manual, there is a glossary that gives simple definitions of common terms. Following the glossary is an index to help you quickly find answers to your questions.

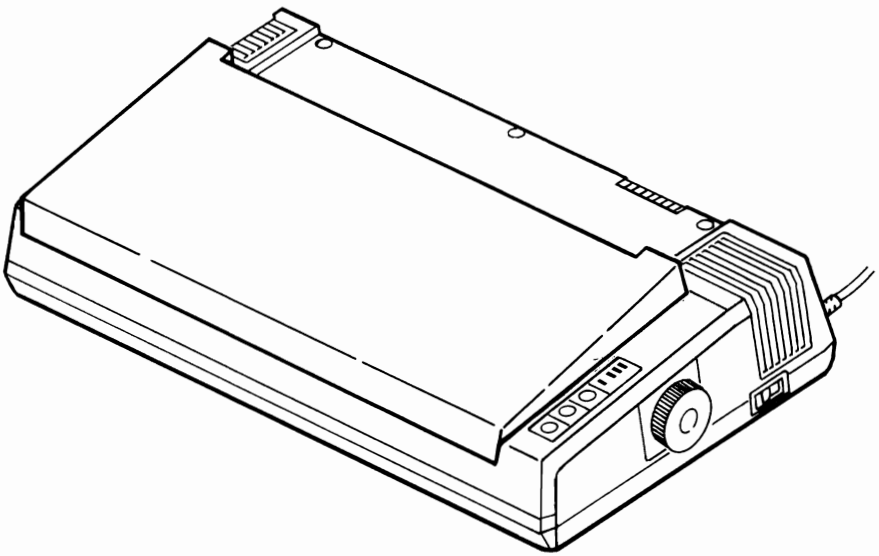
A Quick-Reference Guide is printed on the inside folding flap of the back cover. By reading this manual carefully, and practicing with your Gemini, you should soon be able to achieve full printer performance just by referring to this Quick-Reference Guide.

Congratulations on your purchase of a Gemini. May it work for you long and well.

This User's Manual serves the operators of the Gemini-10X and Gemini-15X printers.



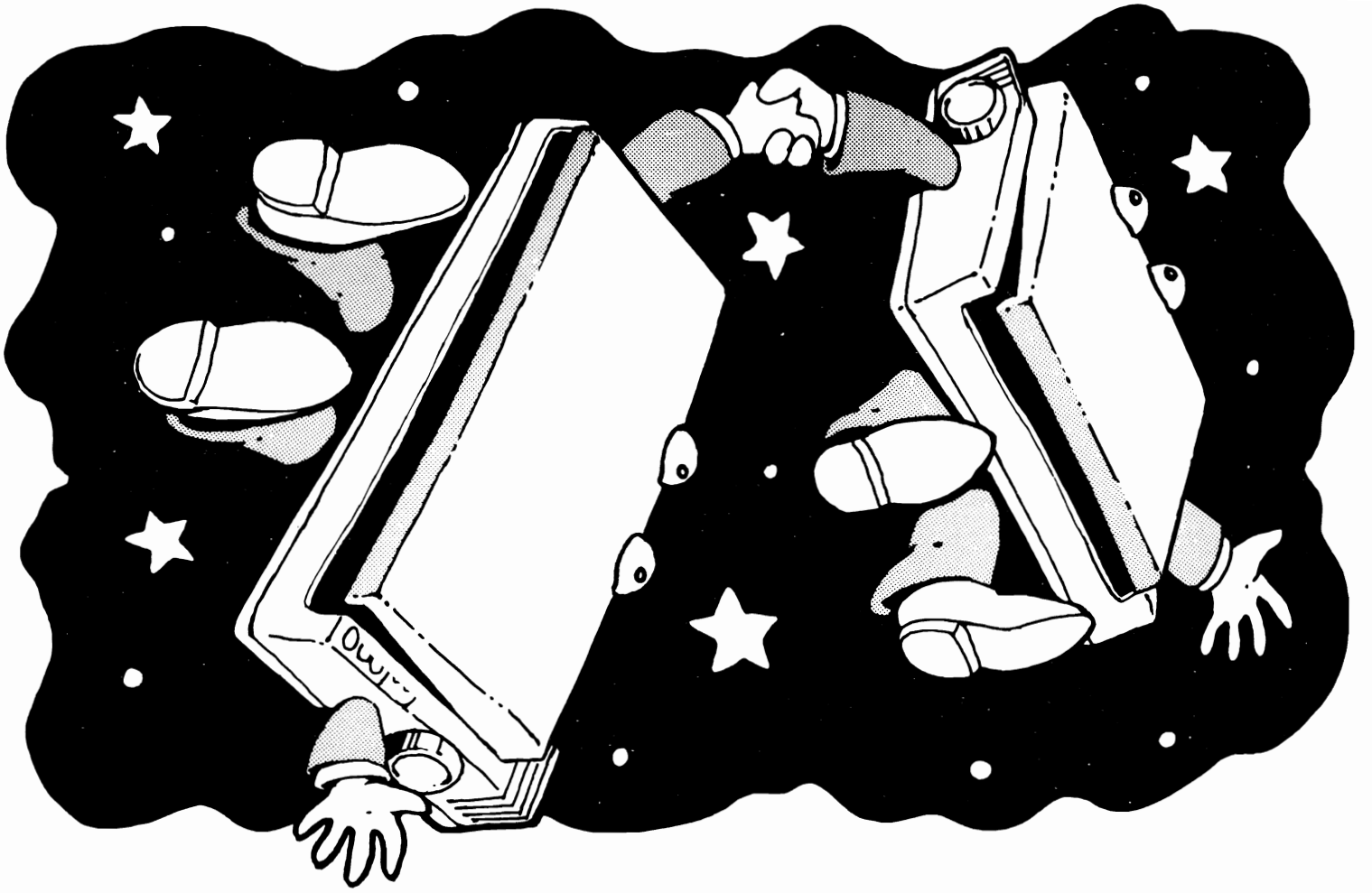
*The Gemini-10X accommodates paper up to 10 inches wide.*

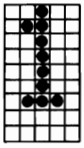


*The Gemini-15X can handle paper up to 15.5 inches wide, and can print more characters per line.*

# *Table of Contents*


| | | |
|---|---|---|
| Chapter 1. | Introduction | 1 |
| Chapter 2. | Paper and Ribbon | 13 |
| Chapter 3. | The Gemini Self-Test | 39 |
| Chapter 4. | Interfacing Gemini with Your Computer | 47 |
| Chapter 5. | Putting Gemini to Work for You | 57 |
| Chapter 6. | The Fundamentals of Dot Matrix Printing | 81 |
| Chapter 7. | The Full Range of Printing Capabilities | 99 |
| Chapter 8. | Downloadable Characters | 115 |
| Chapter 9. | The Function Codes | 123 |
| Chapter 10. | Maintenance | 143 |
| Appendix A. | Apple II Plus | 151 |
| Appendix B. | Atari 800/400 | 165 |
| Appendix C. | Commodore VIC-20, C-64 | 179 |
| Appendix D. | IBM-Personal Computer | 192 |
| Appendix E. | Osborne | 194 |
| Appendix F. | TRS-80 | 206 |
| Appendix G. | When You First Get Your Gemini | 217 |
| Appendix H. | ASCII Code Charts | 226 |
| Appendix I. | ASCII Code Tables | 230 |
| Appendix J. | International Character Sets | 231 |
| Appendix K. | Character Style Tables | 232 |
| Appendix L. | Control Circuit Block Diagram | 241 |
| Appendix M. | Function Code Summary | 242 |
| Appendix N. | Gemini's Parallel Interface | 246 |
| Appendix O. | Code Chart | 250 |
| Glossary | | 254 |
| Index | | 258 |
| Consumer Response | | 261 |
| Regional Service Offices | | 265 |
| FCC Notice | | 266 |
| Warranty | | |
| Reference Guide | | |





# INTRODUCTION

Allow us to introduce you to your Gemini printer: "Gemini, this is your new user." "User, this is the remarkable new Gemini — a versatile and dependable printer that lets you do *tricks* with the dot *matrix* and bit image capabilities."

In this chapter, we'll answer the question: What are the parts of the printer and how do they work together? We'll define the operating environment for Gemini, and we'll discuss its technical specifications. If you are just now unpacking your newly acquired Gemini, **refer to Appendix G to learn how the pieces fit together**, then return to this chapter to discover how to put your printer to work for you.

## A TOUR AROUND GEMINI

Let's begin with a tour around the outside of the Gemini. On the top, right, front part of your machine, you'll see the ***control panel*** (as shown in Figure 1-1). These three buttons and the four status lamps are your day-to-day operational controls. These are fully explained later in this chapter, but for now we'll move on because this is an express tour.

On the right side of your printer is a large knob and, toward the back, you'll find the ***"on/off" power switch***. The knob is attached to the platen (pronounced plat-en), which is the padded roller around which the paper is fed to the printing mechanism. Hence, the handle is called the ***platen knob***, and allows you to manually turn the platen (with the power "off"), just like on a typewriter.

The power switch, located on the lower right side, is the "on/off" switch that controls the flow of electricity to your machine. The Gemini printers use U.S. standard 110-120 volt A.C., found in wall outlets of homes and offices. As a safety precaution, you should make sure that your printer is turned "off" before plugging the three-pronged safety plug into the electrical outlet.

Turning the corner and examining the back of your machine (as shown in Figure 1-2), you'll see the power cord, grounding connector, interface ports, and DIP switches.

The ***DIP switches*** (a popular acronym for the electronic term, "dual in-line package") control important printer functions. Some companies call these "microswitches" because their engineers made them so small you almost need a microscope to find and

set them. Other products call them "internal switches" because you must take their printers apart to get to them. The Gemini DIP switches are easy to set, which is explained in Chapter 4.

Next to the DIP switches is the ***connector*** for the ***parallel interface***. This is where you connect your Gemini to your personal computer, so they are "interfacing," and able to communicate with each other. (Appendices A through F provide information on hooking-up your Gemini with six widely used types of microcomputers.) This connector is for a "parallel" interface, a type of interfacing that is accomplished by having the printer "parallel-wired" like a string of double-wired Christmas lights, where each lightbulb burns independently. Data from your computer is sent down the cable wires to your Gemini side-by-side, or "parallel."

Between the parallel interface and the power cord is the "port hole" (a punch-out plastic tab) for the ***serial interface***. This optional interface allows you to connect Gemini with a computer requiring serial communications.

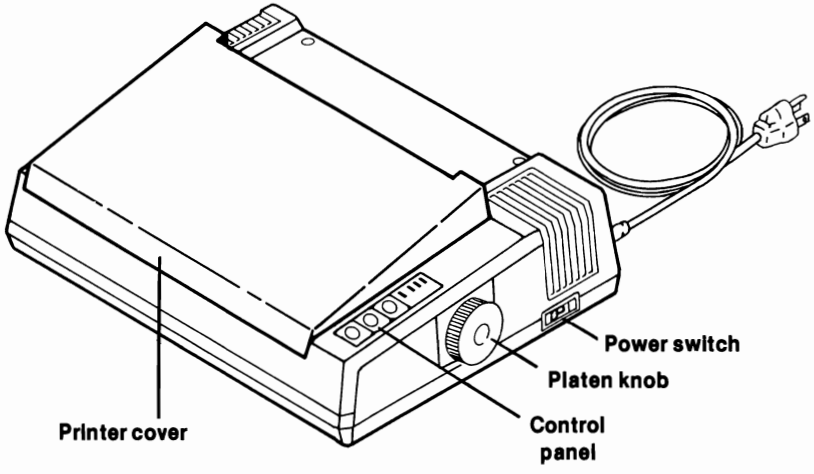



*Figure 1-1. The operational controls are on the right side of the Gemini printers.*

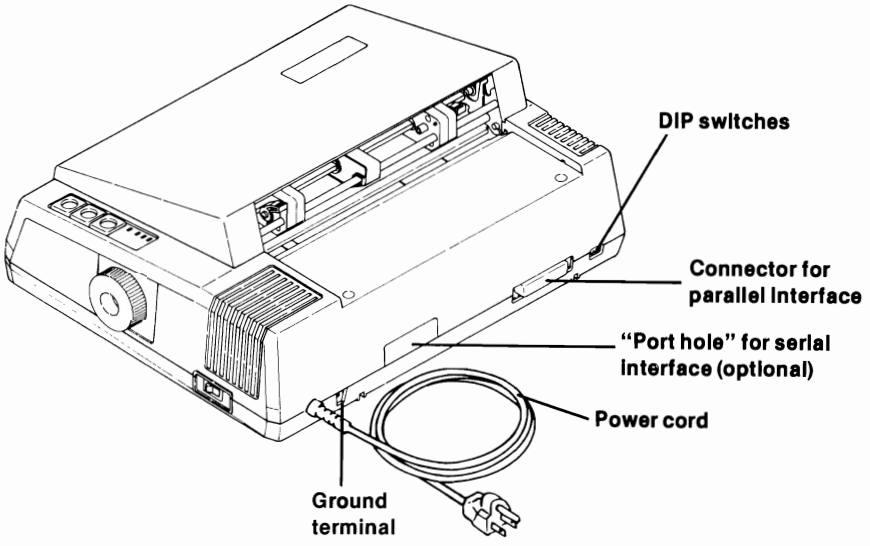



*Figure 1-2. The connectors and DIP switches are located on the back of the Gemini printers.*

## A SNEAK PREVIEW INSIDE GEMINI

By referring to Figure 1-3 we can examine more closely the parts that make your Gemini printer work.

If you are just getting started, you should refer to Appendix G for instructions on unpacking and assembly, then to Chapter 2 for information on loading the ribbon and setting the print head's gap. After you've cleared those hurdles (which you'll do soon enough), this is what your printer will look like.

Inside the front of the machine are two ***ribbon spools***. Gemini uses standard typewriter ribbons, saving you money you might have had to spend for special-purpose ribbon cartridges. When the ribbon has run completely from one spool onto the other, the Gemini automatically reverses the ribbon direction. When the ribbon starts wearing (and the print copy lightens), simply take the ribbon off, turn it upside down, and print on the other edge.

The cylindrical ***print head*** (observable behind and above the ribbon spools) is the ingenious device that can print 120 characters per second (CPS) onto a variety of papers. What's more, the print head is engineered to work consistently for an average of 100 million characters without fail (expressed as 10 x $10^7$ Mean

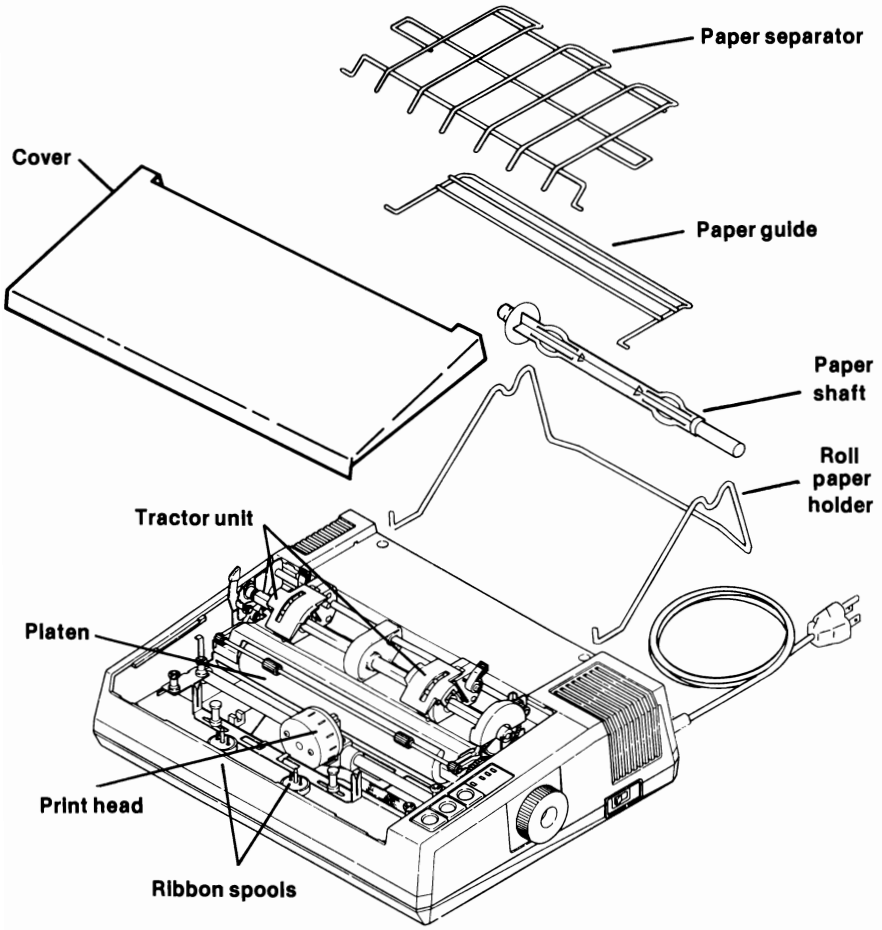


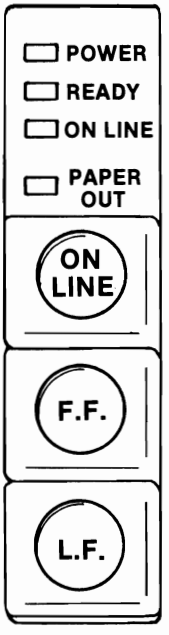

*Figure 1-3. For more information about how the paper guide, separator, and holder should be attached to the printer, please refer to Appendix G at the back of this manual.*

Curve Between Failures). If you ever need to replace the print head, you can detach it yourself for replacement.

The Gemini prints by means of ***pins*** that are thrust from the "working" side of the print head. These pins ***impact*** against the paper, which is backed by the ***platen***. As already mentioned, the platen is like the roller on a typewriter. The individual ***characters*** (letters, numerals, and symbols) that are printed result from the action of the dot-producing pins. These wiry little pins "fire" dots

onto the paper, with the dots arranged according to preset matrix patterns (imaginary grids). Because the characters are printed in a series (one after the other), Gemini is called a ***serial, impact dot matrix printer.***

Above and behind the platen is a device for bearing and propelling fanfold computer paper. The ***tractor*** unit is used with computer paper having perforated and punched paper guides on the sides.




The "power" lamp glows green when the power is "on."

The "ready" lamp gives you the green light for "go" when the printer is ready to accept data. During transmission, the lamp flickers.

The "on line" lamp is green, and glows when data transmission is possible.

The red lamp of "paper out" means just that: your printer is out of paper and has stopped printing.

The "On Line" button lets you alter the printer status between being "local" (which means independent and on its own) and being "on line" (which means able to receive data transmitted from a computer). When you turn the printer's power "on," you're automatically "on line."

F.F. stands for "Form Feed." When you're in the "local" mode, you can tap this button and advance the paper that is being "fed" into the printer to the first line of the next page, or to the start of the next form.

L.F. means "Line Feed" and allows you to advance the paper one line at a time. It's like rolling a page through a typewriter, advancing you to the next line. If you hold this button down, you'll get consecutive line feeds, one right after the other.

*Figure 1-4. The control panel (atop the right front of your printer) tells you the status of your controls, lets you switch between "on line" and "local," as well as feed more paper through your machine. Take Gemini "off line" (so the "On line" lamp goes off) to operate the form feed and the line feed.*

## THE CONTROL PANEL

Let's turn our attention to the buttons and status lamps on the control panel. Figure 1-4 shows what the buttons and lamps can do for you.

The two buttons in front are used to advance paper through the printer, to the next line on which you want to print. ***L.F.*** advances the paper one line at a time, and ***F.F.*** moves the entire form (or page) forward.

The ***Online*** button allows you to switch between being "online" (meaning hooked-up and in a data transfer mode) with your computer, and being "local." ***Local*** means that the printer is *incommunicado* — not in a communicating mode with your computer.

## A PRIMER OF YOUR PRINTER'S BASIC COMPONENTS

This section is provided for those of you who are new to computer printer technology, and for now only want to know the most basic information about how the Gemini works.

Figure 1-5 shows what is called an "exploded" view of Gemini-10X's major components, and Figure 1-6 is an elementary logic block diagram for both models. But, because the Gemini printers are so well designed, it isn't necessary to know detailed information about how it works — just that it does.

Expressed simply, when you turn the power switch "on," the alternating current is converted into direct current for the ***power unit***. The direct current powers the motors and the print head, as well as providing power to the ***control board***. The control board does just that: controls the printer by means of a maze of transistors, resistors, diodes, capacitors, and ***integrated circuits*** (***IC***, for short).

The control board contains the ROMs and RAMs, as explained in the Glossary at the back of this manual. For now, it is enough to know that the control board is the "brains" of the machine. In fact, your new printer is smart enough to realize when it has finished printing a short line, and instead of the print head making an unnecessary trip along the rest of the blank line, Gemini moves the paper up automatically, so no time is wasted before the next characters are printed. This is why Gemini is said to be ***logic-seeking***.

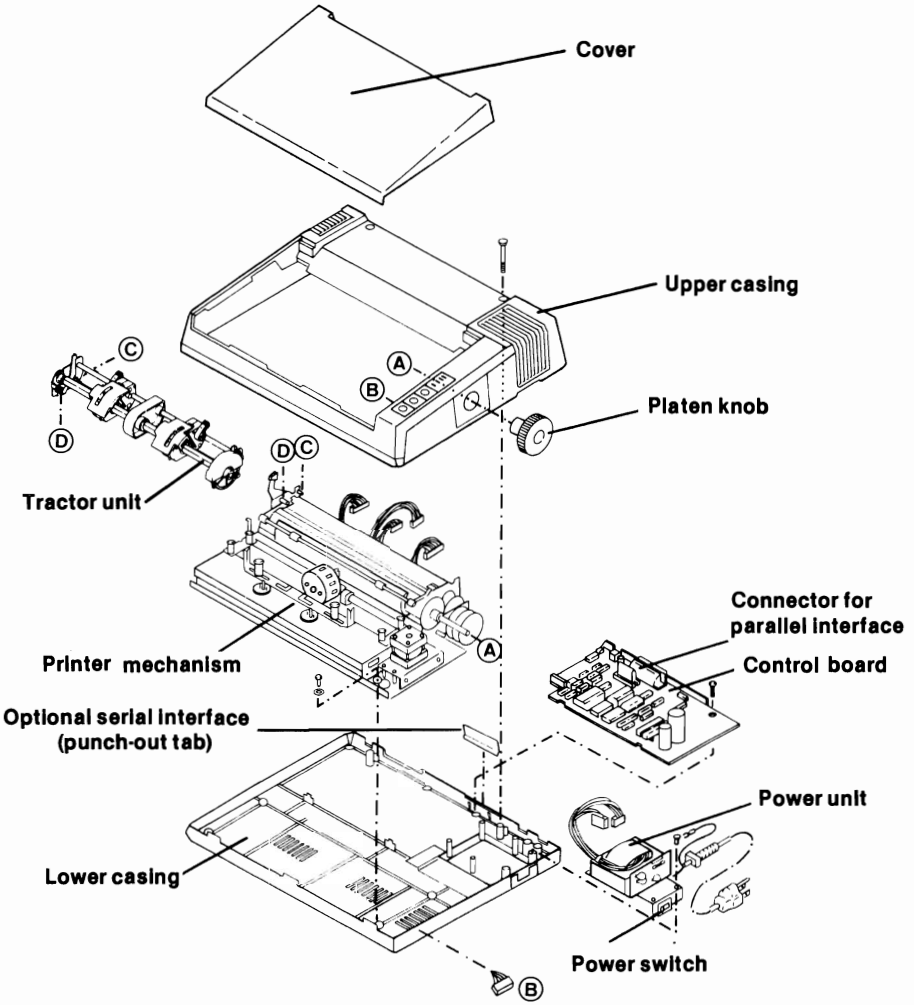



*Figure 1-5. The main components of the Gemini printer are the power unit, the control board unit, and the printer mechanism (which features a detachable tractor).*

The control board also contains the ***buffer*** that allows your computer to send a batch of information to be printed, which Gemini then stores until it can get to it. Although Gemini is a fast

printer, computers communicate data even faster. The buffer can accommodate 816 characters of memory, with an optional buffer that can be expanded to 4K or 8K. *Qu'est-ce que "K"?*

Computer memory is measured in terms of ***K***, meaning ***kilobytes***. A kilobyte is 1,024 bytes, and a ***byte*** equals one letter of the alphabet (or a symbol, or a number).

The print-out is accomplished by the ***printer mechanism***, also shown in Figure 1-5 and represented in Figure 1-6. The printer mechanism can produce a variety of typographical styles and sizes (including $^{\text{superscripts}}$ and $_{\text{subscripts}}$), variable line spacing, adjustable tabs, and block graphics. Plus, the Gemini is outstanding for the full range of papers upon which it can print.

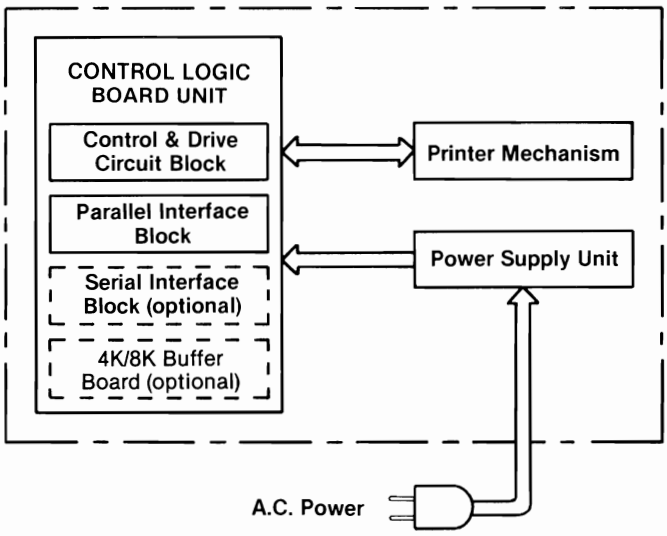



*Figure 1-6. This block diagram presents, in simple terms, the relationship among the transformer, power supply, printer mechanism, and circuit control. For more information on the control circuit, please refer to Appendix L.*

## GEMINI'S OPERATING ENVIRONMENT

When you install your Gemini printer, you should use common sense and good judgment about where you place it. The following are worthy of consideration in evaluating a potential operating environment:

Position Gemini horizontally on a level surface.

Pick a place that does *not* have

- extremes in temperature (very hot or very cold)
- strong vibrations
- dusty atmosphere
- oil present
- metal particles nearby.

Avoid placing Gemini where it will be exposed to direct sunlight, or close to a heater or heat-generating applicances.

Provide a separate power supply, away from noise-generating motors.

The power supply must not be more than 10% more or less than 120 volts A.C. (as specified on the product identification plate). Note: Extremely high or low voltage will cause printer problems.

The Gemini printers are valuable information-handling tools. These tips will help you pick a good setting for the Gemini. Chapter 10 provides maintenance suggestions for keeping your printer in good working order.

## TECHNICAL SPECIFICATIONS

The following table provides a summary of the technical specs for the Gemini-10X and the Gemini-15X. If you understand all of it, great! If you don't, that's okay, too. Either way, by the time you finish reading this manual, you should know more than enough to use your printer to its full potential.

***Note:*** The technical specifications of the Gemini printers are subject to change without notice.

| | |
|---|---|
| Printer: | Serial impact dot matrix |
| Print head: | 9 wire (user replaceable) |
| Print speed: | 120 CPS (at 10 CPI)<br>bidirectional logic seeking. Graphics mode: unidirectional |
| Print buffer: | 816 bytes (expandable up to 4K or 8K) |
| Character matrix: | 9 x 9 standard with true descenders<br>18 x 9 emphasized<br>18 x 18 double strike<br>6 x 6 block graphics<br>60 x 72 low-resolution bit-image graphics<br>120 x 144 high-resolution bit-image graphics<br>240 x 144 ultra-high-resolution bit-image graphics |
| Character sets: | 96 standard ASCII characters<br>96 italic characters<br>64 special characters<br>32 block graphic characters<br>96 Downloadable characters (programmable)<br>88 International characters (selectable) |
| Character fonts: | Normal (10 CPI)<br>Elite (12 CPI)<br>Condensed (17 CPI)<br>Emphasized<br>Double strike<br>Enlarged (5, 6, 8.5 CPI)<br>Super/subscripts<br>Italic |
| Special features: | Self-Test<br>Continuous underline<br>Backspace<br>Vertical, horizontal programmable tabs<br>Left and right margin set<br>7 or 8-bit selectable interface<br>Bit image column scan<br>Perforation skip<br>Downloadable characters (programmable)<br>Macro instruction |

Mean curve between failure (MCBF): $5 \times 10^6$ lines (excluding print head)

Print head MCBF: $10 \times 10^7$ characters

Columns (maximum):

| | GEMINI-10X | GEMINI-15X |
|---|---|---|
| Normal | 80 | 136 |
| Elite | 96 | 163 |
| Condensed | 136 | 233 |
| Enlarged | 40, 48, 68 | 68, 82, 116 |

Line spacing: 1/6, 1/8, 7/72 plus programmable by *n*/72 or *n*/144 inches

Paper slew: 10 lines/second (at 1/6-inch spacing)

Paper handling:

| | GEMINI-10X | GEMINI-15X |
|---|---|---|
| Fanfolded | width = 3-10 inches<br>copies 3 sheets*<br>thickness =<br>0.07-0.1 mm (1 part)<br>0.28 (3 parts) | width = 5-15.5 inches<br>copies 3 sheets*<br>thickness =<br>0.07-0.1 mm (1 part)<br>0.28 (3 parts) |
| Roll paper | width = 8.5 inches<br>thick = 0.07-0.1 mm<br>diameter of roll =<br>5 inches maximum | width = 8.5 inches<br>thick = 0.07-0.1 mm<br>diameter of roll =<br>5 inches maximum |
| Single | width = 8-10 inches<br>thick = 0.07-0.1 mm | width = 8-15 inches<br>thick = 0.07-0.1 mm |

*Note: fanfold = 1 original & 2 carbonless copies

Interface: Standard parallel (Centronics compatible)
Serial RS-232C (optional)

I/F protocol: Parallel by BUSY, ACK
Serial by BUSY, XON/XOFF*, ACK*
*When used in conjunction with the STAR Gemini 10X/15X buffered interface.

Ribbon: Standard Underwood spool-type, 13 x 50 mm

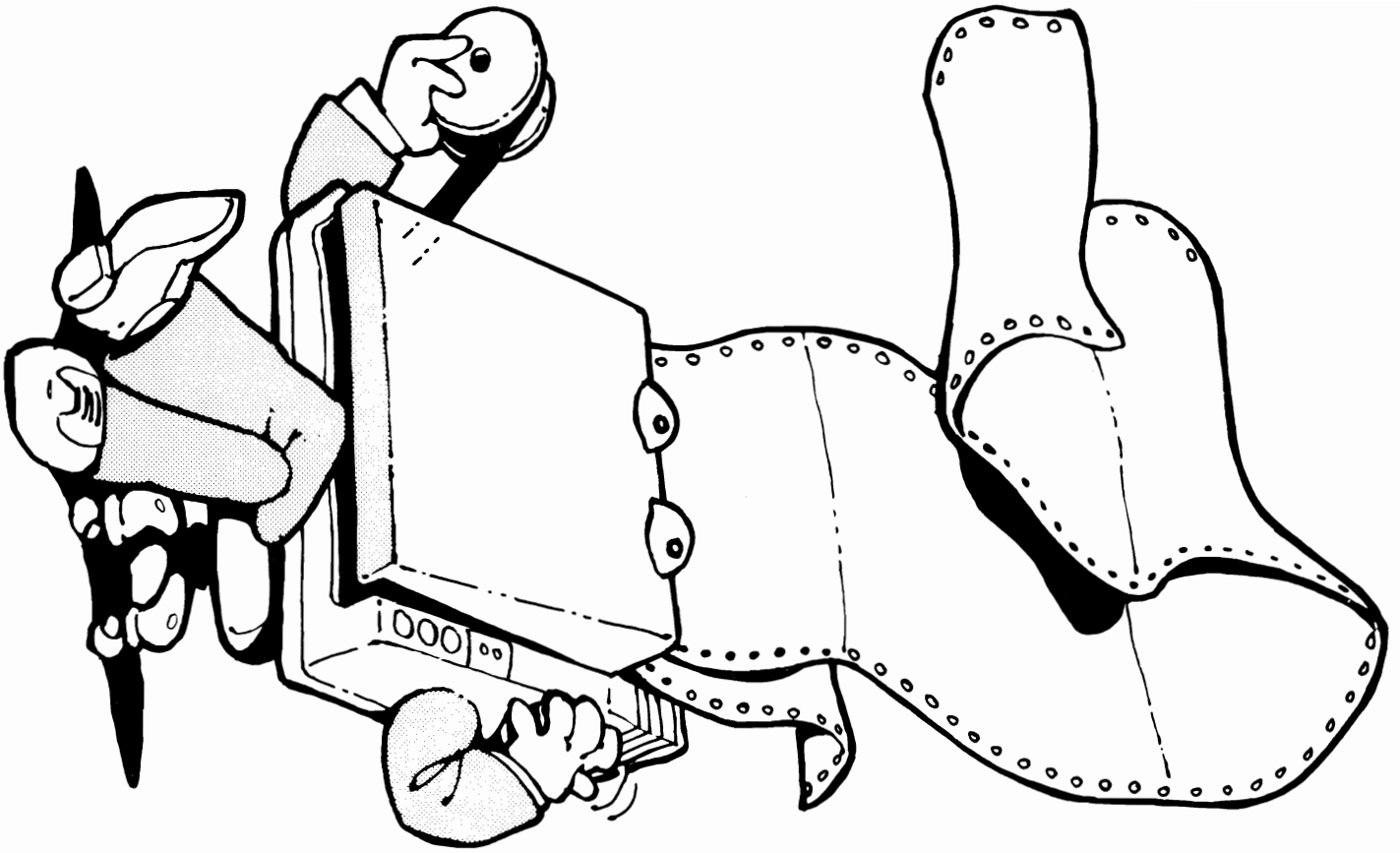
Dimensions:

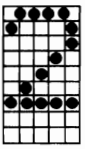
| | GEMINI-10X | GEMINI-15X |
|---|---|---|
| Width | 15.2 inches<br>392 mm | 21.3 inches<br>542 mm |
| Depth | 12.4 inches<br>315 mm | 12.4 inches<br>315 mm |
| Height | 5.8 inches<br>148 mm | 5.8 inches<br>148 mm |

Weight:

| | GEMINI-10X | GEMINI-15X |
|---|---|---|
| pounds | 15.4 | 22 |
| kilograms | 7 | 10 |

Power: 120 VAC ± 10%, 60 Hz., approx. 1A





# PAPER AND RIBBON

With Chapter 1 now "imprinted" in your memory, you're almost ready to apply all these new discoveries about the Gemini printer. In this chapter, we'll learn how to load paper into the printer, install the ribbon, and set the print head gap. Chapter 3 will give you a chance to print something; to see the results of what you've learned so far. But first, let's explore the options for loading different types of paper.

If you've borrowed a Gemini printer (until getting your own), you should refer to Appendix G to learn how to attach the various paper-feed accessories. Appendix G tells you what the new owner gets when first opening the box and how all the pieces fit together. This chapter assumes you know "how" the pieces fit together, and you only need to learn "why."

## LOADING THE PAPER: INDIVIDUAL SHEETS

The Gemini printers are designed to accommodate individual sheets — whether it's standard-size stationery, an off-size memorandum, multilayer carbonless business forms, an envelope, or almost any other single sheet.

The Gemini printers can handle individual sheets of paper according to the following parameters.

**Single-Sheet Size Specifications**

| | Paper Width | | Paper Thickness | |
|---|---|---|---|---|
| Printer | Minimum | Maximum | Mimimum | Maximum |
| Gemini-10X | 8 inches | 10 inches | 0.07 mm | 0.10 mm |
| Gemini-15X | 8 inches | 15 inches | 0.07 mm | 0.10 mm |

## Removing the Tractor Unit

The tractor unit is only used with fanfolded computer paper. (That's the paper that comes in perforated sheets, folded into stacks, with rows of holes along the right and left edges.) To print onto single sheets, you need to remove the tractor unit.

To remove the tractor unit, follow these simple steps:

1. Turn off the printer. (This isn't really a "requirement," but is a good safety precaution, at least until you become more experienced with your Gemini.)

2. Identify the two "snap levers" by comparing your printer with the Gemini-10X shown in Figure 2-1.

3. Pull both "snap levers" forward (one on the right and one on the left of the tractor unit), and at the same time...

4. ...Rock the tractor unit up and back, about a half-inch. The tractor unit rests on two "stoppers" just above the right and left ends of the platen. This step in removing the tractor is done by simply rocking the unit backward after you've released the snap levers that hold it in place.

5. Now lift the tractor unit toward you, up and forward from the printer.

Superman may be able to stop a speeding locomotive, but you just lifted a tractor! With the tractor removed, your Gemini should look like the printer shown in Figure 2-2. (Of course, if you have a Gemini-15X your printer will be wider, but you still get the idea.)

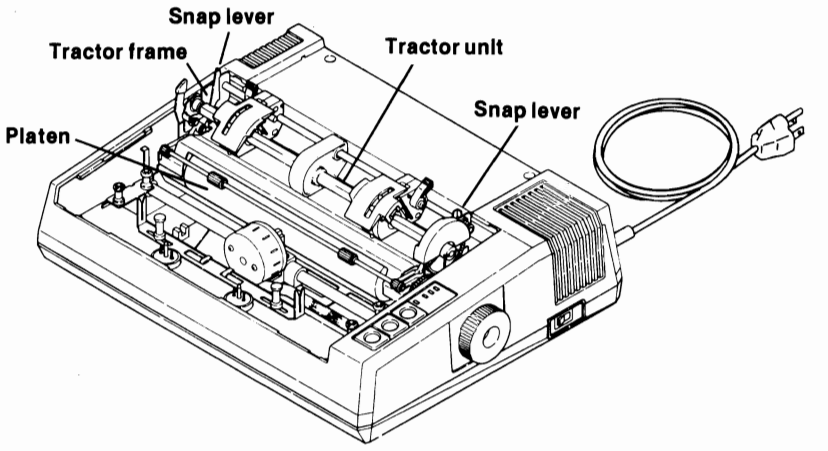



*Figure 2-1. The Gemini tractor unit is easy to remove for using single sheets or roll paper.*

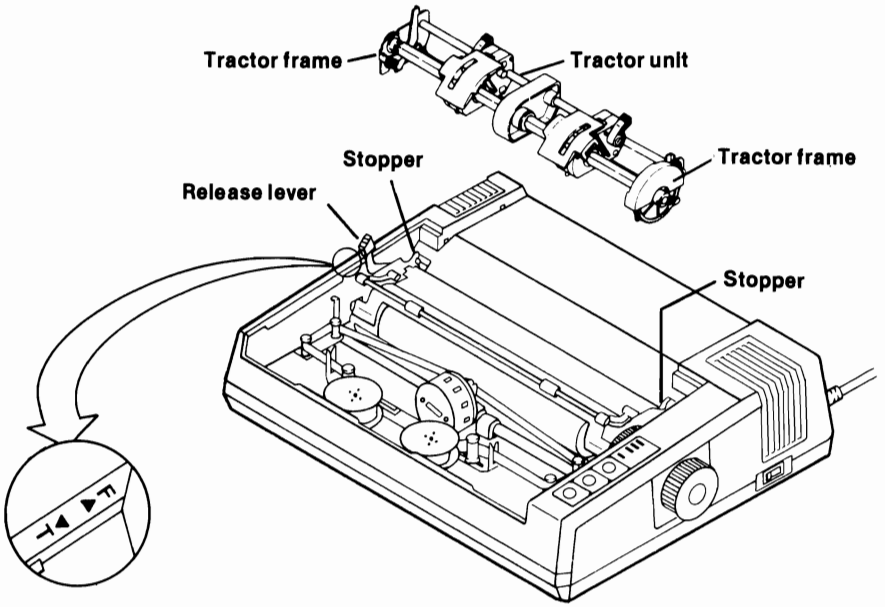



*Figure 2-2. Tractor frames support the unit on the left and the right. Be sure to adjust the release lever when removing or mounting the tractor unit.*

***Note:*** After removing the tractor unit you'll need to set the "release lever" when you are ready to print single sheets. The release lever is located on the left side of the printer (see Figure 2-2). This is somewhat like shifting an off-road vehicle from two-wheel to four-wheel drive. On the Gemini printers, you need to "shift" the release lever forward so it aligns with the "T" for "tractor" when the tractor unit is in use. But with the tractor unit removed, you need to shift the release lever backward so it aligns with the "F." (The "F" stands for "Friction feed," which is the "drive" setting for single-sheet and roll-fed paper.)

## Inserting Single-Sheet Paper

1. Turn the power "off."

2. Make sure the release lever (on the left side of the printer) is set in the "T" position (which is the "open" mode for inserting the paper; you'll "close" it to the "F" mode in a moment).

3. Lift up the paper press, as shown in Figure 2-3. (This step is the same as if you were inserting a sheet of paper into a typewriter.)

4. Insert the sheet of paper from the back side of the platen, between the paper chute and the platen cover. Push the paper through to the upper front side of the platen.

5. Adjust the paper horizontally (right and left) so the sheet is where you want it. Be sure the area you're going to print on the sheet is within the margins you're going to set for the printout.

6. Make sure the paper is in straight.

7. Position the sheet of paper where you want the first line of text to start printing. (It may take you a little practice to get the placement consistently right, but you'll master it quickly.)

8. Push the release lever back to the "closed" position, indicated by "F."

9. Return the paper press to the original position, flush against the paper.

10. Replace the cover and turn the power "on."

***Note:*** Never touch the printing head during printing. Aside from the mechanical injury threat associated with touching a fast-moving device, there is the danger of burning your fingers. As the caution label on the print head states:

**Caution: Hot Surface, Avoid Contact**

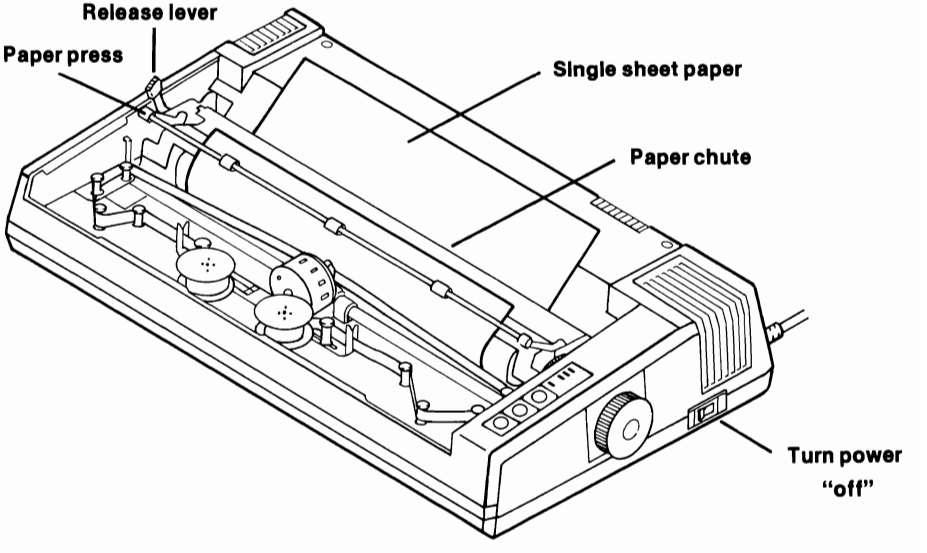



*Figure 2-3. Inserting a single sheet of paper into the Gemini-15X (shown here) and the Gemini-10X is just as easy as rolling a sheet of paper into a standard typewriter.*

## LOADING ROLL PAPER

The Gemini printers also allow you to print from rolls of paper. The specifications are the same for both the Gemini-10X and the Gemini-15X.

Roll paper width: 8.5 inches

Roll paper thickness: 0.07—0.10 mm

Diameter of roll: 5 inches (maximum)

Roll paper, like single sheets, is fed into the printer by what is called "friction feed." Thus, the use of roll paper requires removal of the tractor unit.

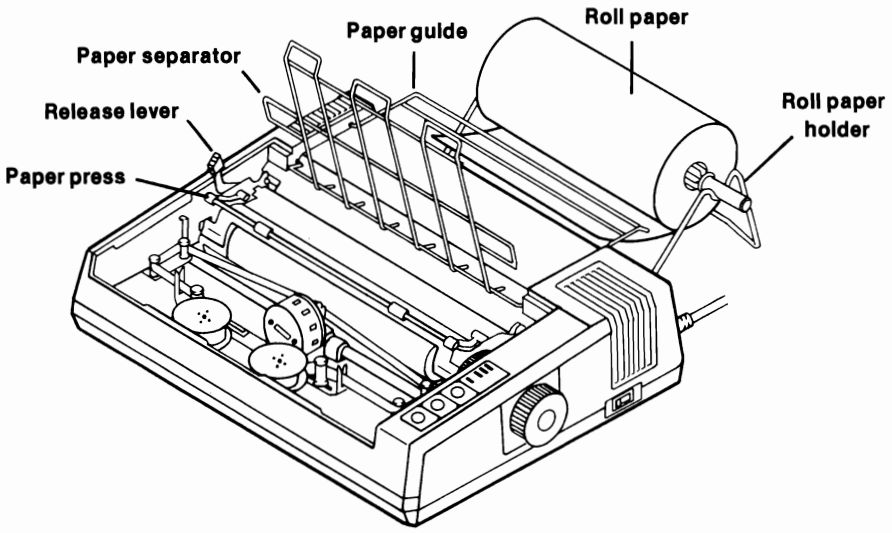



*Figure 2-4. Both Gemini printers use similar paper guides and separators and identical roll-paper assemblies.*

Roll paper utilizes the metal racks, which come as accessories with every Gemini printer. As shown in Figure 2-4, these accessories include the paper guide, paper separator, roll-paper holder, and holder shaft. For information on how to attach these accessories, please refer to Appendix G.

The steps for inserting roll paper are fairly simple and straightforward:

1. Turn the power "off."

2. Remove the tractor unit, if you haven't already done so.

3. Following the procedures described in Appendix G, attach the paper separator, guide, and holder.

4. Pull the release lever (located on the left side of the top-front part of the printer) into the "open" position. As with single-sheet paper, the "open" position is indicated by the letter "T."

5. Lift the paper press.

6. Pull the paper separator upright, as depicted in Figure 2-4 (for the Gemini-10X). The paper guide should remain horizontal.

7. Load the paper roll, as shown in Figure 2-4, so that when viewed from the printer's right side, the paper will unroll in a clockwise direction.

8. As shown in Figure 2-5 (for the Gemini-15X), unroll some paper and pass it above the paper guide and underneath the up-ended paper separator.

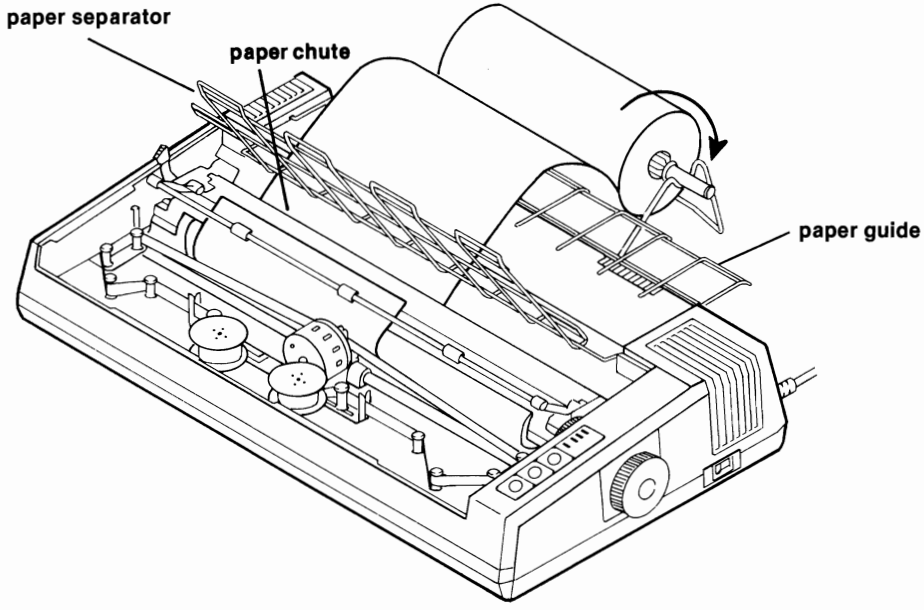



*Figure 2-5. Inserting roll paper in the Gemini printer is similar to inserting single sheets, with accessories to accommodate the fact that the "single-sheet" in this case is quite long.*

9. Insert the roll paper into the paper chute, then push the paper through so it goes around the platen.

10. When the paper has been pushed through so it's above the paper press, push the press back so it's in its operational position (that is, flush against the paper).

11. Adjust the paper, if necessary, so it will feed through straight; then push the release lever back to the "locked" or "closed" position, marked with the letter "F."

12. Lower the paper separator, as shown in Figure 2-6.

13. Rotate the platen knob and check to see if the roll paper is feeding smoothly. If some adjustment is needed, pull the release lever forward to the "open" mode again, and adjust the placement of the paper before "closing" the release lever.

14. Set the print-start position to where you want to begin printing on the roll.

15. Turn the power switch to the "on" position.

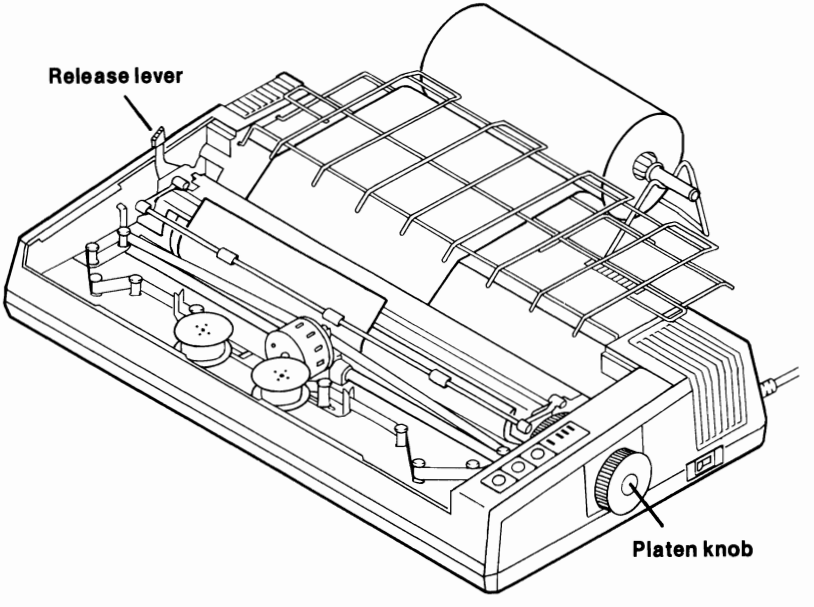



*Figure 2-6. After the roll paper has been inserted and the various mechanisms set in their proper places (steps 10—12), you can turn the platen knob to adjust the roll paper vertically.*

***Note:*** If you encounter any problem finding suitable roll-paper supplies, contact your Gemini dealer.

## LOADING SPROCKET-FED PAPER

Sprocket paper is also known as "computer paper," fanfold paper, and "perf" paper (for "perforated"). Even if you call it "That funny-looking paper with rows of holes on the sides," your Gemini can still print on it.

**Sprocket Paper Specifications**

| Parameter | Gemini-10X | Gemini-15X |
|---|---|---|
| Copy: | | |
| minimum | 1 sheet | 1 sheet |
| maximum | 3-part form* | 3-part form* |
| Paper width: | | |
| minimum | 3 inches | 5 inches |
| maximum | 10 inches | 15.5 inches |
| Paper thickness: | | |
| Single sheets | | |
| minimum | 0.07 mm | 0.07 mm |
| maximum | 0.10 mm | 0.10 mm |
| 3-part forms* | 0.28 mm | 0.28 mm |

**Gemini can print 3-part "carbonless" forms, which consist of one "original" sheet and two "copies."*

### Mounting the Tractor Unit

Figures 2-1 and 2-2 showed you how to remove the tractor unit from your Gemini. The tractor unit drives ("feeds") the sprocket paper through the printer. When you want to print onto this type of paper, you must first attach the tractor unit. Figure 2-7 shows you the correct alignment for gliding the unit into position and snapping it into place.

The tractor units for the Gemini-10X and Gemini-15X are quite similar, but the paper-loading procedures are markedly different. So the following sections instruct you on loading sprocket-fed paper into each type of printer. But first, let's look at the tractor unit and its main parts.

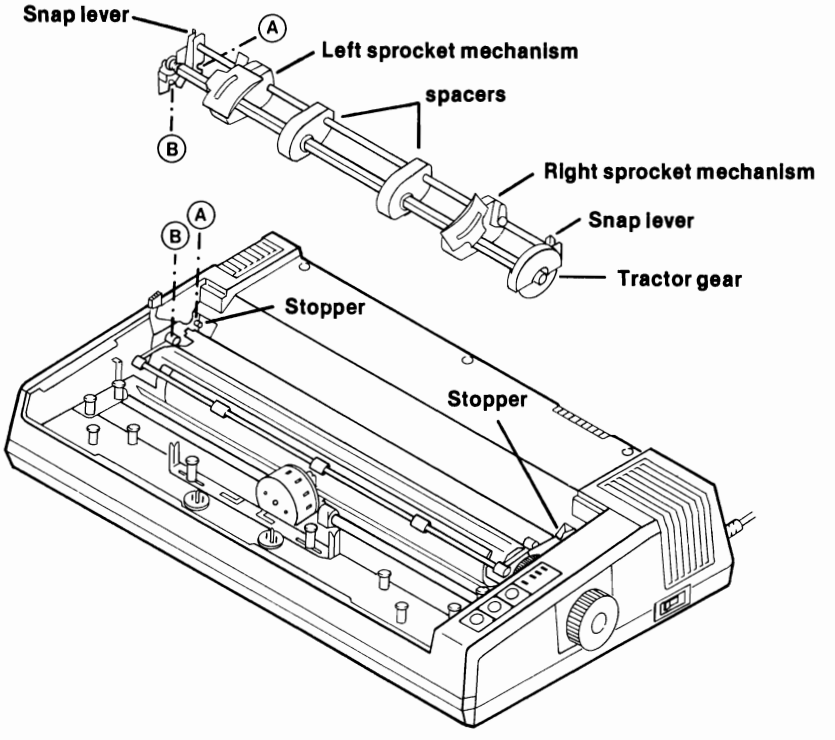



*Figure 2-7. Mounting the tractor unit is a "glide and snap." Glide the unit down until the A's align, then snap the unit down so the B's lock into place.*

Facing your Gemini, as depicted in Figure 2-7, you see the tractor's driving gear-wheel that is on the right (protected by a "fender," complete with a "fender skirt"). To the left of this part is the right sprocket. On the other side of the spacer(s) is the left sprocket unit. These mechanisms drive the paper through the printer. Each has a hood, secured with a spring. Try lifting the hood, if you want.

Toward the back of the top of each sprocket mechanism is a clamp lever. If you snap the lever forward, you can freely move the sprocket mechanisms back and forth to handle different widths of paper. When you have the mechanisms where you want them, snap the clamp lever backward, and it acts like a parking brake.

## Gemini-10X Loading Sprocket Paper

The Gemini-10X should be loaded with sprocket-fed paper according to the following steps.

1. Turn the power switch "off."
2. Install the tractor unit.
3. Install the paper separator and paper guide, as described in Appendix G.
4. Pull the release lever, located on the left side of the printer, toward the "open" position (marked "T").
5. Lift up the paper press.
6. Pull the paper separator upright, as shown in Figure 2-8.
7. Place the stack of fanfolded computer paper behind the printer.
8. Pick up the top sheet of paper and feed it between the paper chute and the platen cover, from the back side of the platen.
9. Push the paper down and forward, so it wraps around the platen.
10. Return the paper separator to its original position.
11. Open the tractor covers atop the right- and left-sprocket mechanisms, as shown in Figure 2-9.
12. Pull the paper up, past the sprocket mechanisms.

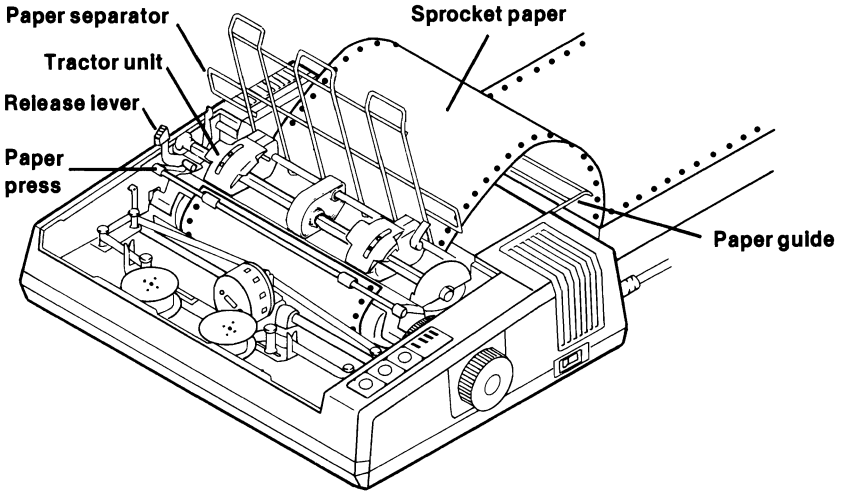



*Figure 2-8. Feed the paper around the platen ("roller") of the Gemini-10X, then push the paper separator "downright." (That's the opposite of "upright" — not to be confused with "downwrong.")*

13. Flip the clamp levers forward on the sprocket mechanisms; adjust each mechanism (right and left) so that the sprockets (the nubby little "teeth" protruding from the sprocket wheel) align with the holes on the paper.

14. Adjust the sprockets so they are lined up with the sprocket holes.

15. Close the tractor covers and snap the clamp levers back into the "locked" position. (See Figure 2-10.)

16. Rotate the platen knob, and roll the paper up or down until the correct "print-start" position is obtained.

17. When the paper installation is completed, replace the printer cover and turn the power on.

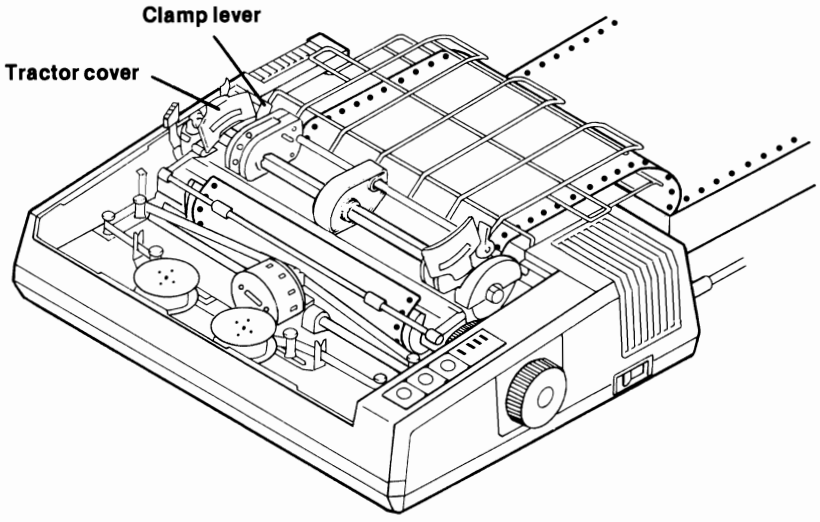



*Figure 2-9. Expose the "teeth" of the sprockets, and align the mechanisms so the paper "feeds" through them.*

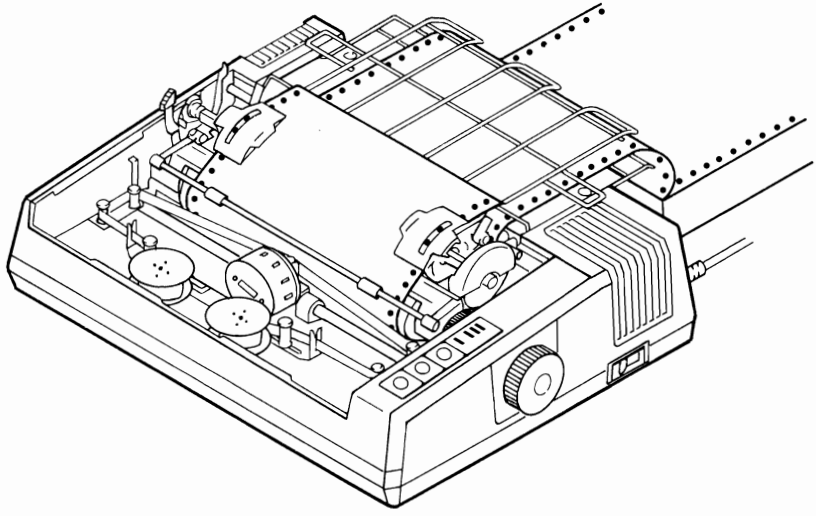


*Figure 2-10. If the sprocket paper you just loaded in your Gemini-10X is multilayer (non-carbon) forms, then refer to the end of this chapter to learn how to adjust the print-gap.*

## Gemini-15X: Loading Sprocket Paper

The Gemini-15X can be loaded with sprocket paper exactly the same as can the Gemini-10X, but with one important difference. The Gemini-15X is designed to accept paper either from the back of the printer or paper fed from beneath the printer. That's right, there's a slot in the undercarriage that lets you position the Gemini-15X above your stack of fanfold computer paper; this paper is fed up through the printer and out the back.

Following are the steps for loading sprocket paper to and through the Gemini-15X. Figure 2-11 illustrates steps 1 through 6.

1. Turn the power "off."

2. Install the tractor unit, as illustrated in Figure 2-7 (on page 23).

3. Attach the paper separator and paper guide (they are the two metal racks that are described in Appendix G).

4. Pull the release lever, located on the left side of the printer, toward the "open" position (marked "T").

5. Raise the paper press.

6. Lift up the paper separator.

7. Raise the tractor covers on the sprocket mechanisms (as shown in Figure 2-12).

8. Snap the clamp levers (atop the sprocket mechanisms) forward to the open position, and move the sprocket mechanisms away from each other until there is enough room between them to accommodate the paper.

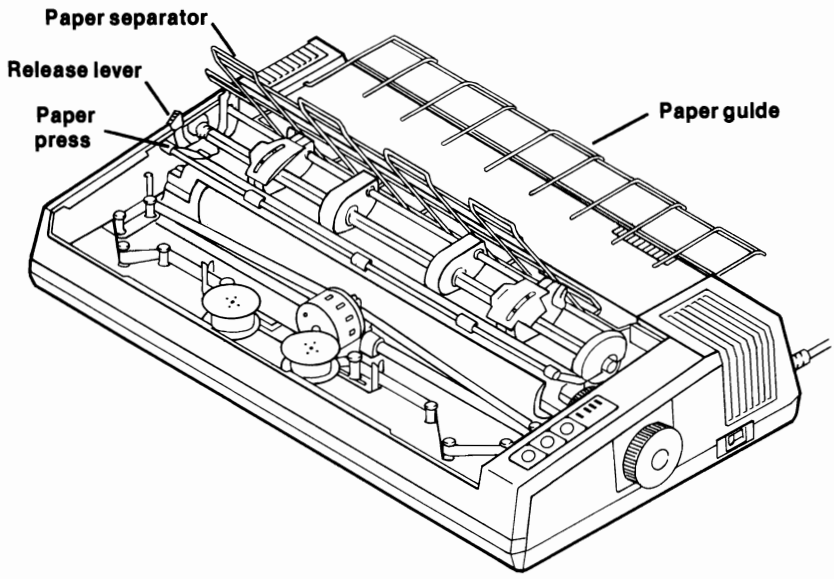



*Figure 2-11. The first step is to prepare the Gemini-15X by putting the mechanical controls in the "open" position.*

***Steps 9—12 lead you down two separate paths:*** Steps 9A through 12A are for loading sprocket paper from "behind" the printer; steps 9B through 12B instruct you on loading paper from "beneath" the Gemini-15X.

9A. Place the stack of fanfolded computer paper behind the printer.

10A. Insert the sprocket paper from the back of the printer (as shown in Figure 2-12).

11A. Feed the top sheet between the paper chute and the platen cover, and around the platen to the front.

12A. Push the paper up past the paper press, and return the paper separator to its horizontal position.

The other option is to:

9B. Place the sprocket paper below the printer (ideally, on the second shelf of a special-purpose printer table, with the printer above it on a shelf with a built-in slot).

10B. Pick up the first sheet of the perforated stack of fanfolded pages, and lift it up and through the slot in the undercarriage of the Gemini-15X.

11B. Push the paper up toward the front of the platen.

12B. Feed the top sheet inside the paper press and past the platen, far enough up so you can grip the paper from above the printer.

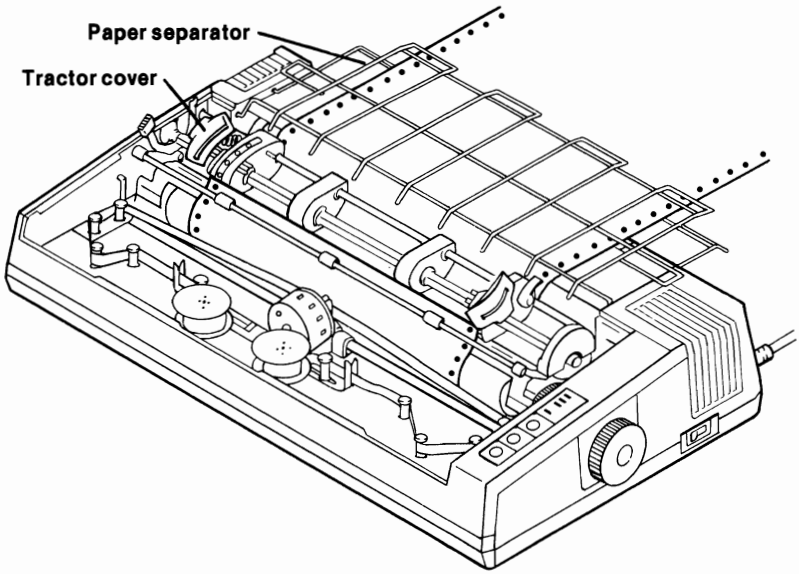



*Figure 2-12. The Gemini-15X can be loaded with sprocket paper from the back (as shown here) or from a stack of paper beneath the printer (depicted in Figure 2-13).*

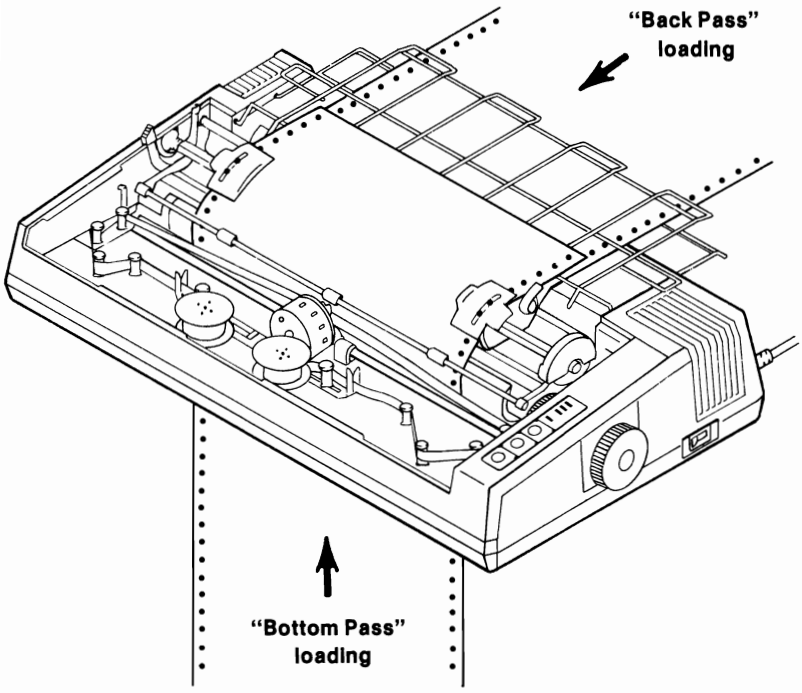



*Figure 2-13. Whether you use the "bottom pass" (shown here) or the "back pass" option (also shown), you start and end with the same steps.*

13. Adjust the paper so that it will feed through straight, and push the paper press back to its operational position.

14. Horizontally adjust the sprocket mechanisms right or left so they align with the sprocket holes on the paper.

15. Vertically turn the sprocket "teeth" (the little nubs on the sprocket wheels) so they line up with the holes on the paper.

16. Snap the clamp levers back into the "closed" position.

17. Close the tractor covers.

18. Rotate the platen knob, and set the print-start position of the paper.

19. Turn the power "on" and put your Gemini to work for you.

***Note:*** The best "start-position" (step 18) will vary according to the paper on which you are printing and the number of blank lines in your document. If you are a novice with dot matrix printers, you might try to position your paper just above the in-place paper press for your first attempts. With practice, you'll be able to place the paper right where you want to start printing.

***Another note:*** If you use multipart paper (such as a three-sheet carbonless form) with your Gemini-15X, you'll need to adjust the print gap, as discussed later in this chapter.

**Congratulations! You've learned how to load the paper.**

Next, we'll discuss the ribbon and print gap,
then you'll be on your way to less reading and more printing.

## LOADING THE RIBBON

You already know that your Gemini printer is one of the best buys available today in terms of what you get for your initial investment. What's more, Gemini's sensible ribbon requirements are even more economical as time passes.

That's because Gemini uses ordinary typewriter ribbons on spools. When the ribbon has gone from one spool to the other, Gemini automatically reverses the ribbon direction.

After the ribbon starts wearing out (when your printed material starts getting too light), you can invert the ribbon and start over.

Replacing the ribbon (which you'll have to do eventually) is easy and inexpensive.

**Ink Ribbon Specifications**

| | |
|---|---|
| Ribbon Color | black |
| Material | nylon (#40) |
| Dimensions<br>width:<br>length: | <br>1/2 inch (13 mm)<br>11.5 yards (10.5 m) |
| Spool Standard | Underwood's plastic type<br>(13 x 50 mm diameter) |
| Recommended Product | Type SF-02B made by<br>Fuji Kagakushi Kogyo |

Near each end of the ribbon, there is a metal or plastic eyelet that signals Gemini to reverse the ribbon direction. It will do this automatically. When loading the ribbon, you should be careful to place the eyelet properly in the ribbon's threading configuration.

## How to Set the Ribbon

Telling you how to set the ribbon is like writing a narrative describing how to tie your shoelaces. Figure 2-14 and 2-15 show you the way to thread the ribbon on the Gemini-10X and Gemini-15X, respectively. But before you try it, you would do well to review the hints provided here.

*Hint #1:* Turn the power "off."

*Hint #2:* Slide the print head gently with your fingers to the approximate center of its pathway (Figures 2-14 and 2-15).

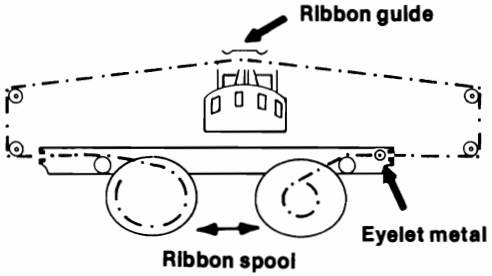



*Figure 2-14. The ink ribbon pass for the Gemini-10X.*

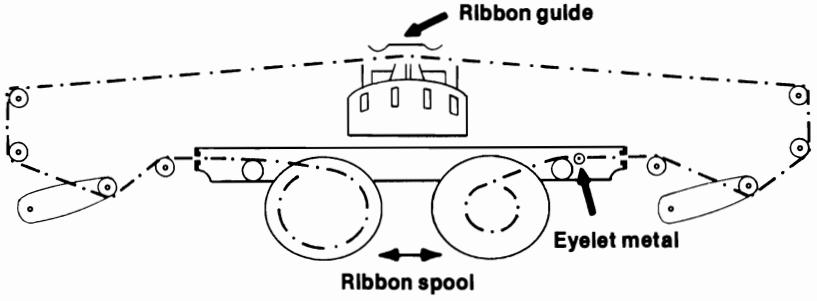



*Figure 2-15. The ink ribbon pass for the Gemini-15X.*

*Hint #3:* Begin by setting either ribbon spool securely on a ribbon spool axle. Make sure you have the spool positioned so it will wind/unwind as shown in the figures. Place the spool securely on the axle, so that the "drive pins" engage the spool.

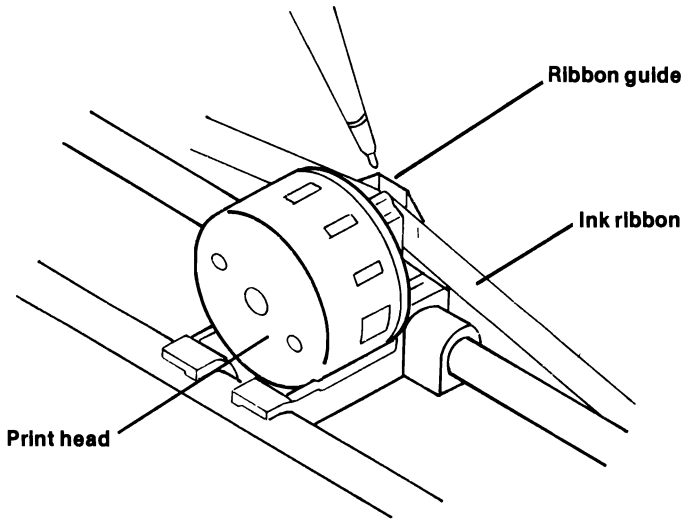



*Figure 2-16. You might want to use a ball point pen to lightly press the ribbon guide toward the platen while inserting the ribbon.*

*Hint #4:* Then thread the ribbon, taking care around the print head. Figure 2-16 shows how to insert the ribbon in the slot between the print head and the ribbon guide. Figure 2-17 shows you how to position the ribbon to avoid misprints or to prevent the ribbon coming off during printing.

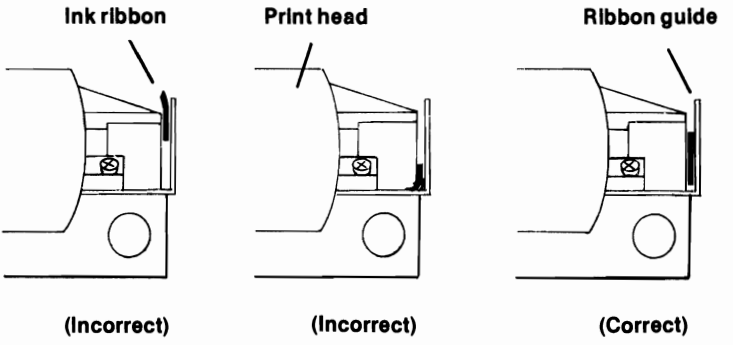



*Figure 2-17. Two wrongs don't make a right, so try to be careful when setting the ribbon near the print head (shown here).*

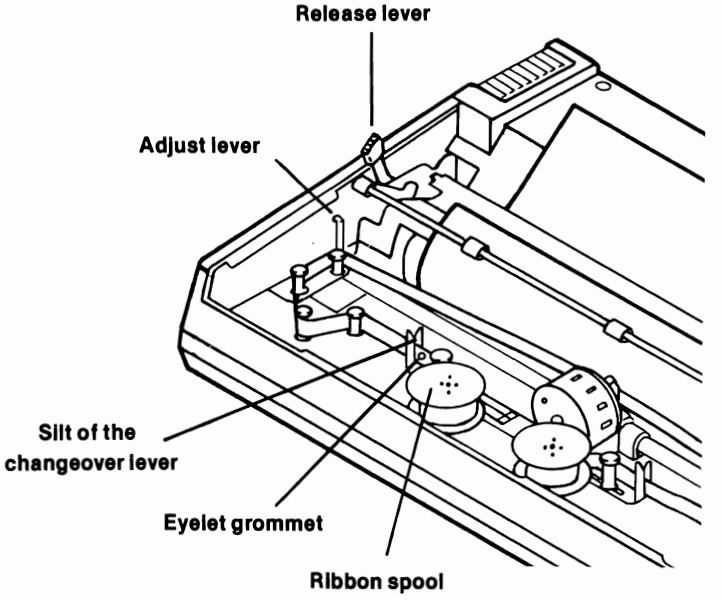



*Figure 2-18. The eyelet grommet (which signals Gemini to automatically reverse the ribbon direction) is shown correctly placed on the ribbon-spool side of the changeover lever.*

*Hint #5:* The eyelet grommet (which signals Gemini to automatically reverse the ribbon direction) should be placed on the ribbon-spool side of the changeover lever.

*Hint #6:* Set the other ribbon spool on the opposite spool axle, making sure the "drive pins" are engaged.

*Hint #7:* When the ribbon threading has been accomplished, turn the ink ribbon spools by hand, approximately four or five rotations of the spools. This verifies that everything was done properly and that the ribbon is smoothly fed.

*Hint #8:* As shown in Figure 2-19, you can turn the ribbon over and use the other side when it starts wearing out.

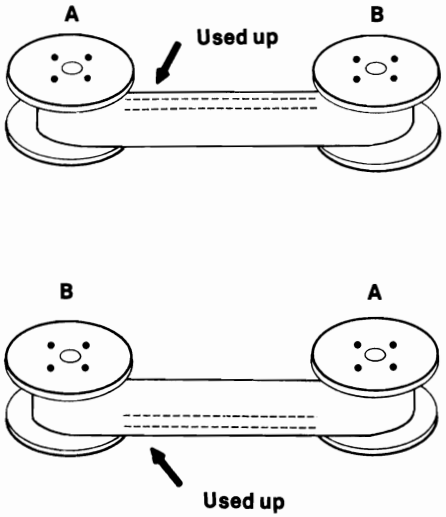



*Figure 2-19. Turning the ribbon over gives you the same effect as having a new ribbon but at no extra cost.*

## ADJUSTING THE GAP BETWEEN THE PRINT HEAD AND PLATEN

Gemini printers can accommodate papers of varying thicknesses by allowing you to change the print-head gap. This "gap" is the distance between the print head and the platen.

Gap adjustment is done by moving the "adjust lever," which is immediately in front of the "release lever" (see Figure 2-18). Pulling the "adjust lever" forward will widen the gap, and pushing it backward will make the print-head gap narrower.

There are five positions that can be attained by moving the lever to different adjustments. These five positions can be changed by the control arm rest in different notches, as shown in Figure 2-20. The "second position" is most commonly used for single sheets of paper.

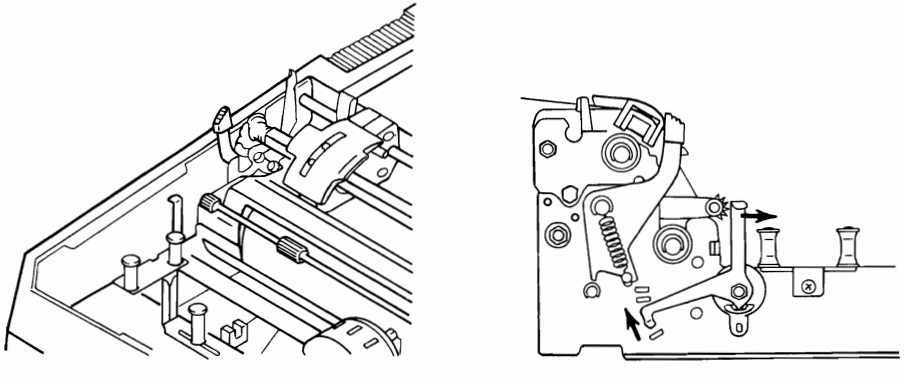
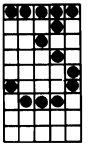


*Figure 2-20. The different positions of the print-head gap-adjustment lever allow you to print on paper ranging from 0.07 mm (that's thin) to 0.28 mm (for three-layer carbonless forms).*

If you are unsure of which is exactly the right print-head gap adjustment to make, experiment. In time, you'll get the best results.

With the turn of a page, you'll be ready to check your work, as Chapter 3 introduces you to Gemini's Self-Test.

REPORT
CARD



# GEMINI SELF-TEST

"Self-Test" is the name of the built-in program that prints out several sample lines of letters, numbers, and other characters to show you that everything is in good working order. You can print the Self-Test without hooking up your Gemini to a computer.

The Gemini Self-Test is a special feature that allows you to check your installation of the paper and ribbon and the adjustment of the print head gap. At the same time, the printout provides a preview of what's to come: all the standard characters in both normal and italic type styles.

## HOW TO INITIATE THE SELF-TEST

The Gemini Self-Test is as easy as "one, two, three:"

1. Plug the printer's electrical plug into an electrical outlet.

2. Insert a sheet of paper.

3. While holding down the "LF" button on the control panel, turn the power switch "on."

The Self-Test (shown on the following two pages) will print out automatically. If you hold down the "LF" button, the Self-Test will continue printing out.

The Gemini-10X Self-Test (Figure 3-1, a & b) consists of four lines that show the standard character set that your printer can produce. The Gemini-15X Self-Test (Figure 3-2, a & b) uses the same characters, and is also displayed on four lines.

After you run the Self-Test on your printer, you can skip to Chapter 4 if you are in a hurry to hook-up your Gemini to your computer and start printing out data that you generated. On the other hand, if you want to take a few minutes to finish this chapter, you'll learn what these characters represent and how to use them.

## WHAT THE SELF-TEST LOOKS LIKE

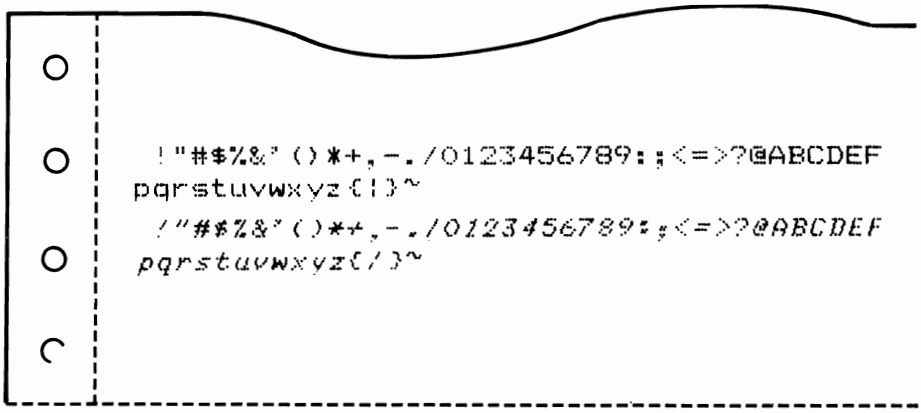



*Figure 3-1a. The left side of the Gemini-10X Self-Test printout (shown actual size).*

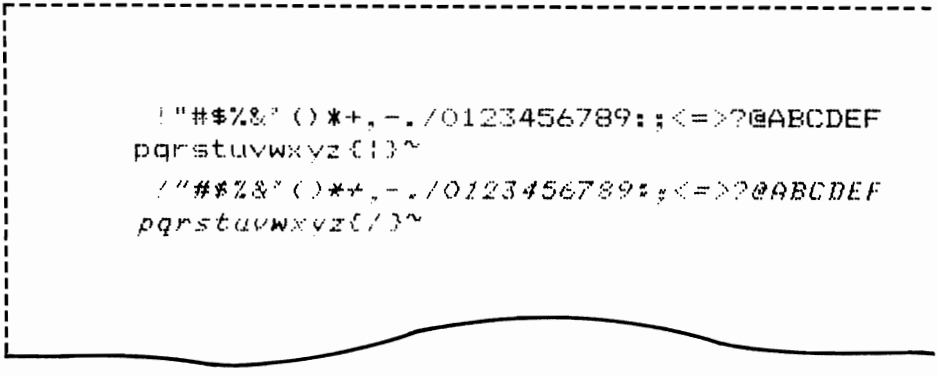



*Figure 3-2a. The left side of the Gemini-15X Self-Test printout (shown actual size).*

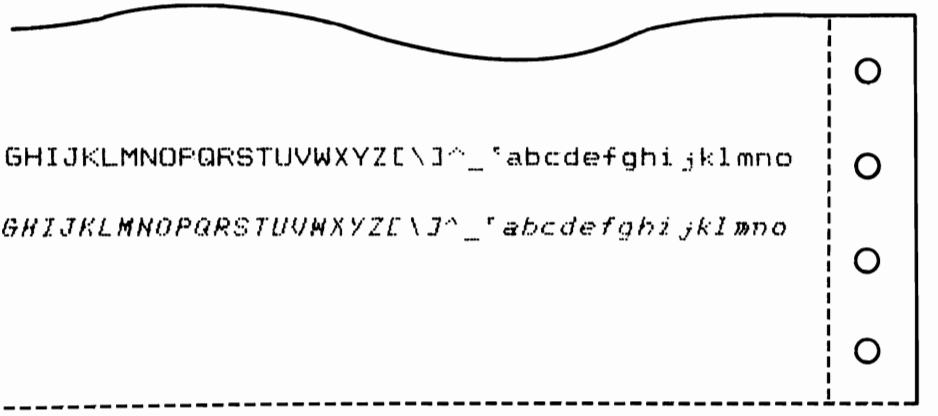



*Figure 3-1b. The right side of the Gemini-10X Self-Test, which is a continuation of this two-part figure.*

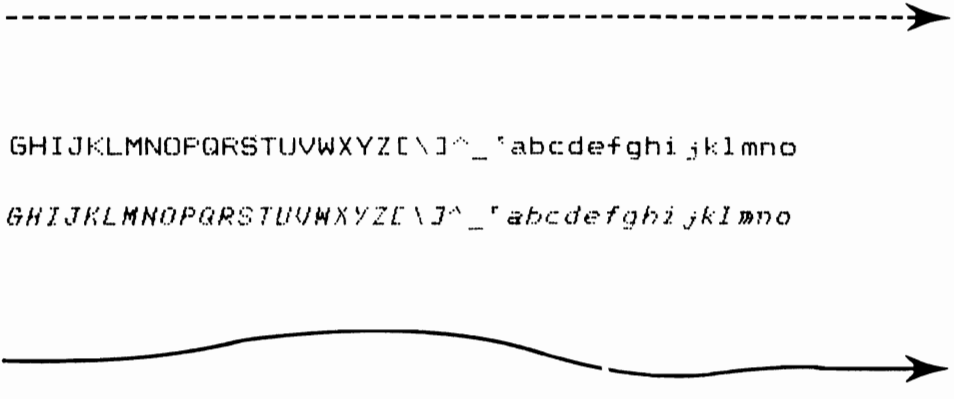



*Figure 3-2b. The right side of the Gemini-15X Self-Test, which is a continuation of this two-part figure.*

## WHAT THE SELF-TEST MEANS

The Gemini Self-Test allows you to test the printer's performance before putting it to work. Beyond that, the Self-Test is a demonstration of some of the characters Gemini has stored in its permanent memory.

On the following pages, groups of characters from the Self-Test are identified and discussed. Where a character has multiple meanings or uses, this manual provides the most common and generally used.

In discussing these characters, we face the old problem of defining a word without using that word in the definition. To make it easier to identify which character we're discussing, we've coded some of the characters to correspond with letters of the English alphabet. If you look at Figure 3-3, you'll see that the fifth character shown in the sequence on the lower line is the "percent" sign. For purposes of our discussion, we've coded it so it corresponds to the letter "f" above it. In Chapter Five, you'll learn about other "codes" that have been created to facilitate discussion about computer-generated characters.

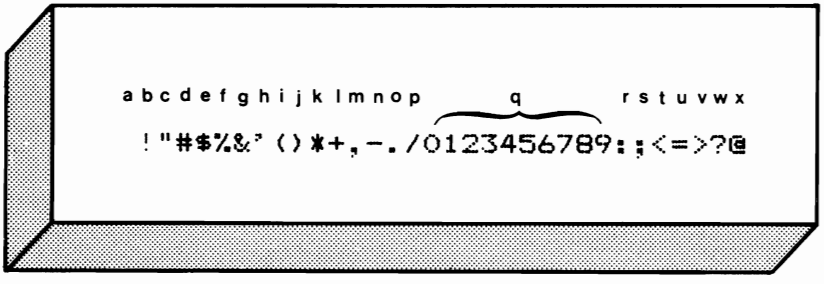



*Figure 3-3. The first part (of the first Self-Test lines) shows the characters often associated with the top row of keys on a typewriter.*

In Figure 3-3, the letter "a" codes a character that's as hard to see as the wind. That's because the first line begins with a blank space.

The "blank" character may not seem important, but without it:

everythingwouldruntogetherlikethis

The first character that appears in the lower line of Figure 3-3 is the "exclamation point" (coded "b"). This is also called the "factorial" sign in mathematical symbolism.

The next eight characters (represented by "c" through "j") are, respectively: double quotation marks, the "number" sign (if it precedes a numeral: as in #5; or a "pound" sign if it follows: as in 5#), the dollar sign, the percent sign, the ampersand (also called the "and" sign), the apostrophe (which also serves as the "final single quotation mark), and the opening and closing parentheses.

The next six characters in Figure 3-3 (represented by letters "k" through "p," respectively) are: the asterisk (or star), the "plus" sign, a comma, the "minus" sign (which also serves as both the hyphen and the short dash), a period, and the "slash" mark. The "slash" (represented by letter "p" is also called the "diagonal," "slant," "solidus," and "virgule"; in the English money system, it represents "shillings."

Letter "q" in Figure 3-3 represents the 10 characters for the numerals "zero" through "nine." It might help to note the difference between a "number" and a "numeral" at this time. A "byte" can represent a character (such as the letter "m" or the numeral "5"), or a whole number. A "number" is a concept, and a "numeral" is a character or group of characters that represent the concept.

In Figure 3-3, the letters "r" and "s" stand for the "colon" and "semicolon" characters. The next three symbols are mathematical signs that show the relationships of values. "<" means "is less than" (such as "3<4"); "=" is the "equals" sign (and the "double-bond" sign in chemistry); and ">" means "is greater than" (such as "6>4"). Next comes the "question" mark, and finally the "at" sign. In everyday usage, "@" may be used like this: "4 diskettes @ $5 = $20." In some software packages, such as Visicalc, the "at" sign is part of a powerful command sequence.

Figure 3-4 presents the second part of the first line in the Gemini Self-Test. The first 26 characters are the English alphabet's capital letters. These are also called the "upper case" letters.

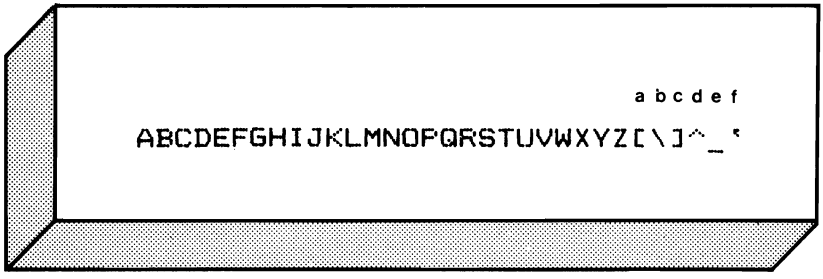



*Figure 3-4. The Self-Test prints the uppercase alphabet in the first line, 10 characters per inch. In Chapter 5, you'll learn how to print larger and smaller letters (from 5 to 17 characters per inch).*

In Figure 3-4, the characters coded "a" and "c" are "brackets." These are sometimes used in the same manner as parentheses. The character coded "b" is a "reverse slash" — the mirror-image of the "slash" in Figure 3-3. The next character, "^," is a computer symbol for depressing the "control key" on your keyboard. It can also be used for a circumflex or exponentiation. Completing this set of characters, the character coded "e" is the "underlining" for a single character, and "f" represents the "opening single quotation" mark.

In Figure 3-5, the "lowercase" equivalents of the "uppercase" letters in Figure 3-4 are presented. These letters are also 10 characters per inch (CPI) in the Self-Test. Following the lowercase letters in this figure are four characters coded "a" through "d." The first and third are "braces," which provide an alternative to parentheses and brackets, especially in grouping mathematical terms. The symbol coded "b" is a special scientific symbol, to which you may assign meaning according to its use. Finally, the character marked "d" is the tilde sign or diacritical mark.

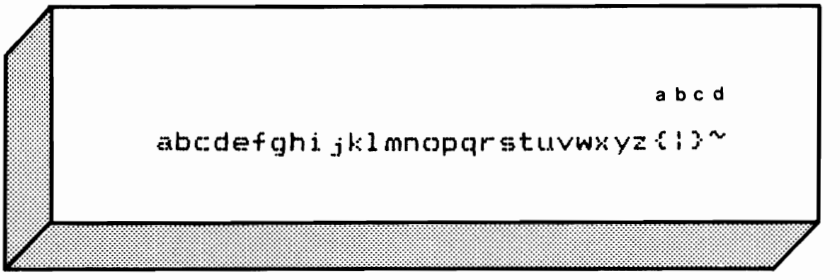



*Figure 3-5. This segment of the Gemini Self-Test shows you what the lowercase letters look like, unless you command the printer to use another type style or a different size.*

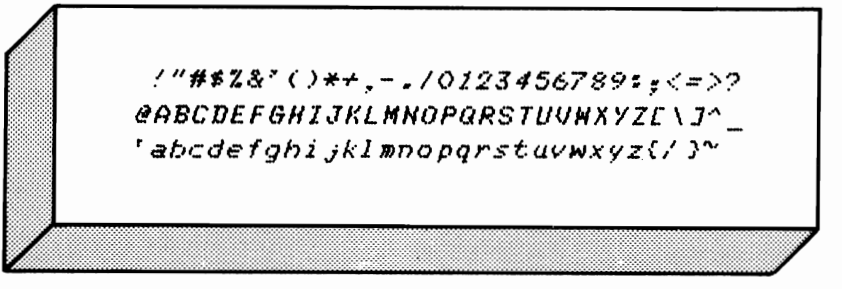


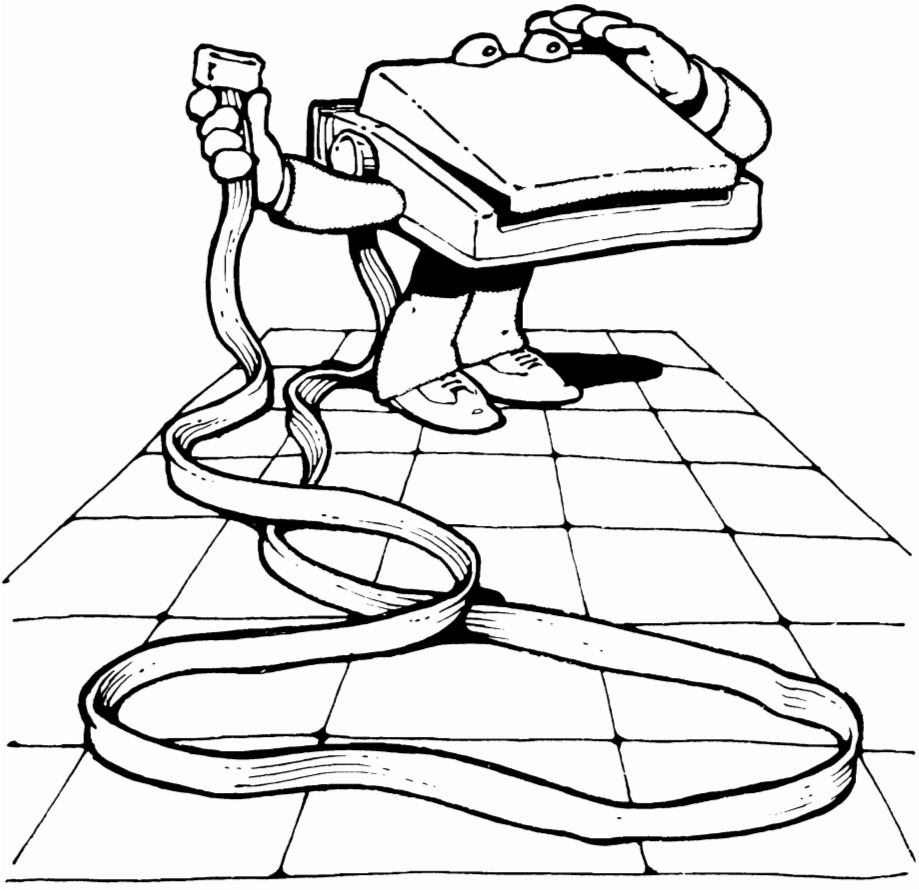

*Figure 3-6. The 96 italic characters form the third line of the Gemini-15X Self-Test; the third and fourth lines with the Gemini-10X.*

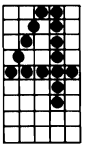
These characters are the italic version of the characters individually identified in Figures 3-3, 3-4, and 3-5.

## TAKE A CLOSER LOOK

Now that you've considered the 96 "standard" characters and the 96 italic characters, you're ready to hook your Gemini up to your computer and get started.

If you wonder what these characters would look like if viewed up close, you might want to take a moment and turn to Appendix K.





# INTERFACING GEMINI WITH YOUR COMPUTER

Your Gemini printer is a remarkable example of modern engineering. A few short years ago, the information-handling capability built into Gemini would qualify the "printer" to be considered a "computer" of sorts. When Gemini is ***interfaced*** ("hooked up to") your microcomputer, the results can be dazzling.

The characters identified in Chapter 3 are imbedded in Gemini's ***read-only memory (ROM)*** chips. Your computer doesn't need to tell Gemini *how* to print the set of dots that represents a certain character; your computer just needs to tell Gemini *which ones* and *where.*

When your computer identifies the characters selected for printing, Gemini stores up to 816 bytes of data in its buffer at the same time that it's printing. Gemini even knows when it runs out of paper — stopping the printing process and (if you wish) signaling you of the problem.

Before we explore interfacing Gemini with your computer, let's briefly consider how Gemini's control circuit is configured. As graphically illustrated in Apendix L (control circuit block diagram), Gemini features a ***central processing unit (CPU).*** This CPU controls the electromechanical operations, such as the print head, carriage motor and paper feed motor. This CPU utilizes the ***random-access memory (RAM).*** It receives data serially, or in parallel, depending on which type of interface your system requires.

## INTERFACE OPTIONS

Gemini provides a parallel interface as standard. Parallel interface is common to almost all popular microcomputers currently on the market. But some computers use serial processing transmission and require serial interface capability. As shown in Figure 4-1, Gemini is designed to accommodate either type of interface.

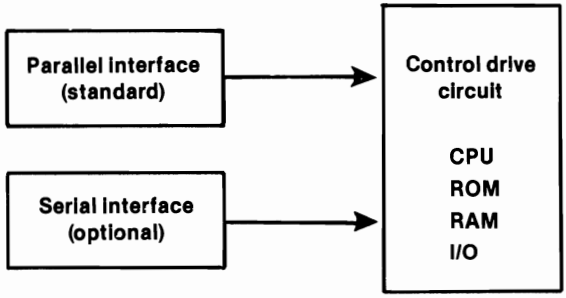



*Figure 4-1. The Gemini printers are provided with the parallel interface as standard equipment, with a serial interface available as an option.*

The parallel interface is incorporated in the same ***printed circuit board (PCB)*** as the ***control circuit board.*** The optional serial interface board is constructed to be attached directly to the PCB and to fully reside within the body of the printer. If your system requires a serial interface, contact your Gemini dealer for purchase of the ***Gemini SBI-4010X serial interface board.***

In the following section, we will discuss the parallel interface. But first, let's consider the DIP switches — where are they, what are they, and what do they do? So turn your printer around, and let's look at the back.

## The DIP Switches

The dual-in-line package (DIP) switch is a device that controls the printer's "patterns of thought." Several functions can be selected by appropriately setting the two DIP switches located on the control logic board (Figure 4-2). DIP switch 2 can be set at the rear of the printer as shown in Figure 4-3. It is necessary to remove the upper casing to set DIP switch 1. Please refer to Chapter 10 for instructions to remove the upper case. A status chart has been provided on the back panel of this manual to assist you in remembering the setting of each DIP switch in your printer.

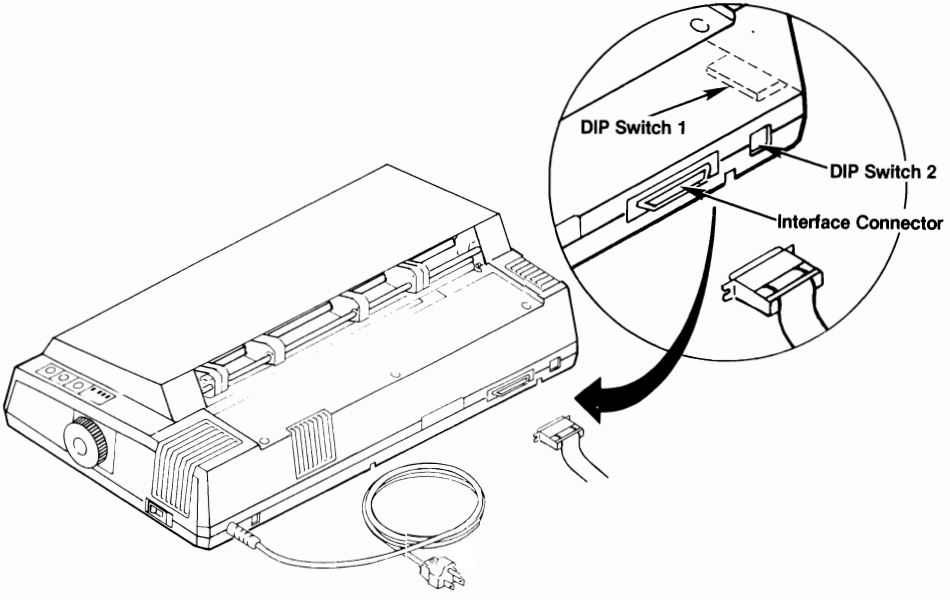



*Figure 4-2. DIP switches on control logic board.*

***Note:*** Always set the DIP switches with the printer's power turned "off."

Each pin in the DIP switch serves a distinct and separate control function. The functions of DIP switch 1 are defined in Figure 4-4. Figure 4-5 illustrates DIP switch settings for selection of the various character sets. Figure 4-6 illustrates the settings of DIP switch 1-6, 1-7, and 1-8 to choose international character sets. Figure 4-7 contains the DIP switch 2 functions.

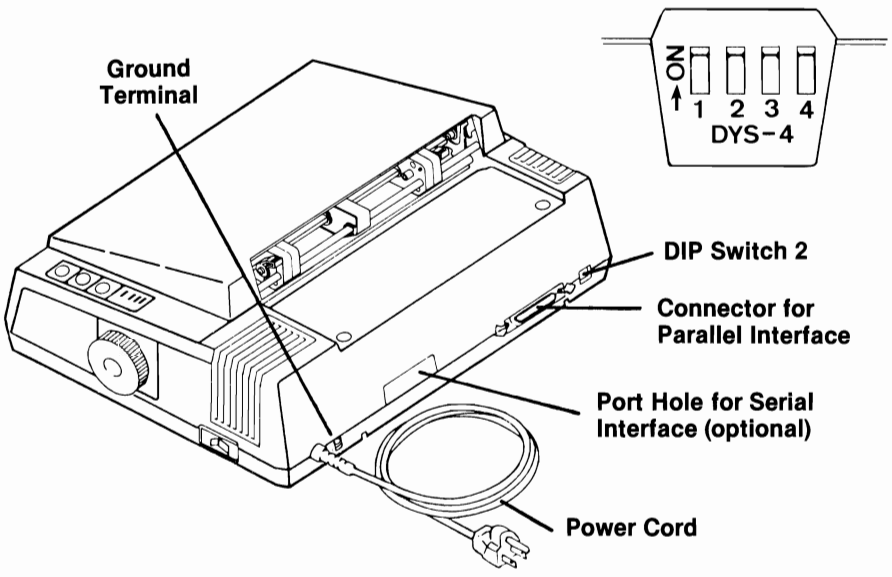



*Figure 4-3. DIP switch 2 controls (1) detection of when your Gemini runs out of paper, (2) the effect of the CR codes on buffer-full printing, (3) 7-bit or 8-bit interface, and (4) automatic line feed.*

| Dip Switch Pin No. | Function | When "On" | When "Off" | Factory Setting |
|---|---|---|---|---|
| 1-1 | Selection of Character Set | (see Figure 4-5) | | on |
| 1-2 | Print mode at Power-On | Normal | Emphasized | on |
| 1-3 | Set of Print Pitch | 10 CPI | 17 CPI | on |
| 1-4 | Selection of Character Set | (See Figure 4-5) | | on |
| 1-5 | Line Feed Value | 1/6 inch | 1/8 inch | on |
| 1-6 | Selection of International Character Set and Form Length | (see Figure 4-6) | | on |
| 1-7 | | | | on |
| 1-8 | | | | on |

*Figure 4-4. The eight switch settings and functions for DIP switch 1. Note that switch 1-1 and 1-4, as well as 1-6 through 1-8 are explained in Figure 4-5 and 4-6 respectively.*

| If DIP Switch 1-1 and 1-4 are set as follows: | | Then, your Gemini will print the following character sets |
|---|---|---|
| 1-1 | 1-4 | |
| on | on | Standard ASCII |
| on | off | Download |
| off | on | Italic ASCII |
| off | off | Italic ASCII |

*Figure 4-5. The setting of DIP switches 1-1 and 1-4 control the character sets.*

| DIP Switch Settings | | | Country | Form Length |
|---|---|---|---|---|
| 1-6 | 1-7 | 1-8 | | |
| on | on | on | U.S.A. | 11 inches |
| off | on | on | England | 11 inches |
| on | off | on | Germany | 12 inches |
| off | off | on | Denmark | 12 inches |
| on | on | off | France | 12 inches |
| off | on | off | Sweden | 12 inches |
| on | off | off | Italy | 12 inches |
| off | off | off | Spain | 12 inches |

*Figure 4-6. By setting different combinations of DIP switches 1-6, 1-7, and 1-8, you can select different international character sets and the form lengths that are standard in those countries.*

| Switch | Function | When "on" | When "off" |
|---|---|---|---|
| 2-1 | No-paper detection | Gemini signals when you are out of paper | The "out-of-paper" signal is not sent by Gemini enabling the printer to print with no paper. |
| 2-2 | Buffer full | The CR code is ignored, and one line of print data is stored until memory is full or the LF code is input | Printing is performed each time the CR code is input |
| 2-3 | Selection of the number of bits | 7-bit interface | 8-bit interface |
| 2-4 | Automatic line feed | CR code automatically performs line feed | No line feed is performed by input of CR code |

*Figure 4-7. DIP switch 2 is "on/off" toggles for four important printer operations.*

The "on" position is when the toggle is flipped "up"; the switch is turned "off" when the toggle is pointed "down." The switches are numbered on Gemini, with the left-most toggle identified as "no. 1" when viewed from the rear and above.

Setting the DIP switches involves using the point of a ball-point pen (or a similar tool that can serve as a small "finger") by gently flipping each switch either up or down.

Appendices A through F respond to the need for setting the DIP switches for six popular types of home computers. Please refer to these appendices for information about the recommended settings for your computer. If your computer is not listed in these appendices, consult your computer's user manual or your Gemini dealer.

Depending on the type of system you have, DIP switch number 2-3 is an important DIP switch that affects computer performance. For example, Apple operates on a 7-bit interface, and the IBM-PC requires an 8-bit interface.

## CONNECTING THE PARALLEL INTERFACE

This section discusses hooking up your computer to Gemini, via the parallel interface.

### A Look at the Parallel Interface Connector

When viewing your Gemini, as shown in Figure 4-3, the parallel interface connector is just to the left of DIP switch 2. Take a moment to look closely at the connection. You will see that the connector is a metal band encasing a plastic spacer, with the center taking the form of a hollow rectangle. This rectangular opening is bordered both on the top and bottom with rows of 18 metal strips, called ***pins***, that are set in the blue plastic spacer.

For more information about the pins in Gemini's parallel interface, turn to Appendix N at the back of this manual.

Each of these 36 pins supports an important function in your computer/printer interface, so the connecting plug must be compatible with the Gemini pin configuration.

### The Connecting Plug and Cable

***Note:*** The cable length should be as short as is workable to minimize signal loss.

Star Micronics, Inc. recommends a connecting plug by ***AMPHENOL*** — product no. ***57-30360***.

The connecting cable should be a twisted-pair, ribbon-type cable. Appendices A through F provide information on acceptable connecting cables, including ***pin-out charts***, in case you want to make your own connecting cable.

## Interfacing Gemini with Your Computer

***Warning:*** Power to your computer and printer must be "**off**" before attaching the interface cable.

The parallel interface connector on Gemini is a rounded trapezoid, with the top edge longer than the lower edge. The connecting plug must be complementary with the Gemini connector; otherwise, it won't fit at all. Figure 4-8 shows the connection and the clamp pins for securing the union.

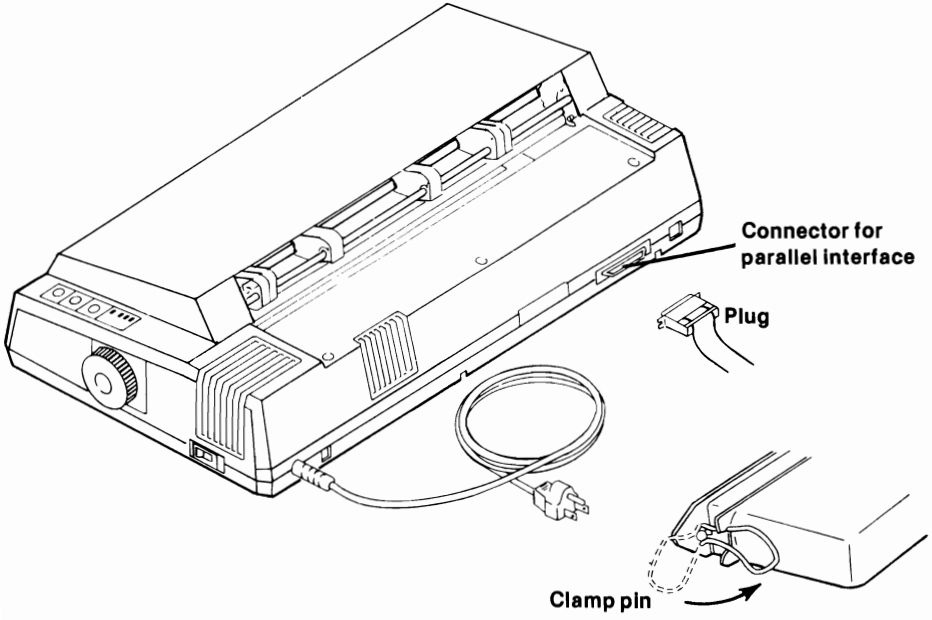



*Figure 4-8. Securely attach both the clamp pins to ensure a good electrical connection.*

Attach the other end of the connecting cable to your computer in accordance with the specifications for your microcomputer. If you are uncertain of what the connector should look like, consult your computer's user's manual.

## SERIAL INTERFACE

The serial interface requirements vary with the type of connector and type of device to which Gemini is interfaced. The serial interface option can be purchased from your Gemini dealer and is accompanied by its own documentation.

## INITIALIZATION

In Chapter 5, you'll get a quick course in putting Gemini to work for you. But the first step is ***initializing*** the printer for operation.

This can be accomplished in three ways. The easiest method is to turn the power switch "off," then turn it back "on" again. Another way is to apply the $\overline{\text{INPUT PRIME}}$ signal (parallel pin no. 31) to the interface connector (see Appendix "N"). The third method is by transmitting the ESC @ code to Gemini, as discussed in Chapter 5 and 9.

Initializing Gemini is somewhat like "cleaning off the blackboard" to get ready for the next printing assignment. All special settings are cleared away and the "standard" character size, line height, and number of lines per page are restored. Altogether, six operations are accomplished when Gemini is initialized.

1. The print head goes to its "home" position (except in the case of ESC @).

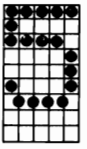
2. The buffer contents are cleared.

3. The characters in download character RAM are cleared (except in the case of ESC @).

4. Character pitch is set to 10 CPI or 17 CPI according to DIP switch 1-3.

5. Line pitch is set to 1/6 inch or 1/8 inch according to the DIP switch 1-5.

6. International characters and form length are set according to the DIP switches 1-6, 1-7 and 1-8.



# PUTTING GEMINI TO WORK FOR YOU

Your Gemini printer is an extraordinary machine. You can use it as a "super" typewriter, a graphics terminal, and as a regular computer line printer. Using standard routines on your computer, you can produce hard copy of program listings, computer memory dumps, and program outputs (text and graphics).

Gemini's standard features include variable-pitch characters (that is, different sizes), adjustable line spacing, left and right margin control, settable tabs, italic type, and special character symbols. High-resolution graphics can be created using the bit image column scan mode.

In this chapter, you will get an accelerated course in how to produce most of Gemini's special effects. If you stop reading this manual at the end of this chapter, you should know enough to be able to dazzle your friends with your command of the Gemini. So let's get started.

## TALKING TO YOUR PRINTER

When you type on a conventional typewriter, your fingers tell the machine which characters to print by striking different keys. You can control the margins, tabs, and spacing of the lines by setting various mechanical controls. If there are removable-type "elements" on your electric typewriter, you can change the type style to italic or special symbols (such as math or Greek).

Aside from the DIP switches and control panel buttons atop Gemini, there are no controls to set on your printer to produce various printing effects. Everything is controlled from your computer by means of electrical signals carried via the connector cable. In addition to telling the printer which characters you want to print, we send Gemini coded messages that change the type size and style, set tabs, vary the line spacing, etc.

In Chapter 3, we discussed the Self-Test and used a "letter-code" corresponding to the individual printed characters. That simple type of coding may have worked there, but another (more sophisticated) coding is required for your computer to efficiently communicate with your Gemini.

## The ASCII Code

Every letter, numeral, and other character that is sent from the computer to Gemini is transmitted in the form of a number code. A ***standard code*** was agreed upon to allow greater compatibility among the hundreds of different types of computers and peripherals. This industry-wide uniform code is called the American Standard Code for Information Interchange, or ASCII (pronounced "ask-key")

Please turn to Appendix K and you will see each character printed by Gemini and a corresponding ASCII code. The decimal number "65" is the code representing the uppercase letter "A." The lowercase "a" is represented by "97." For every character, there is a corresponding code number.

The ASCII code numbers for the uppercase letters, numerals, and punctuation marks are fairly well standardized. Thus, most computers share the same ASCII codes between 32 and 127. Unfortunately, the rest of the codes (128 through 255) are nowhere near standardization in all computer systems.

***Note:*** This is a timely opportunity for you to review the manual that accompanied your computer and to compare your computer's ASCII character codes with Gemini's. Somewhere in the back of your computer manual, you should find a table or chart of ASCII character codes. Compare the chart with Appendix H at the back of this manual. *Mark the characters that are different,* for future reference.

## Function Codes

Besides letters and numerals, you can send special commands to functionally control printing operations. These codes are identified variously as "function," "control," and "command" codes, but they serve the same purpose. In this manual, we call them function codes and they are listed in Appendix M and in the ***Quick-Reference*** chart (on this manual's inside back cover).

Chapter 9 explains what these codes accomplish. For now, you need only note that some codes are repetitions of others. Because not all computers can send all the ASCII codes, Gemini accepts any of several function codes to cause the same effect.

## Transmitting the Codes

The easiest way to transmit functional control codes is to build them into the software program that sends the regular codes for letters and numerals. Sending the control codes to the printer by building them into the program allows you to change the printing instructions during the program. This gives you total control to print what you want, the way you want to print it.

You can also send control codes to the printer using the computer system's "command level" functions before running a program.

All of this discussion about sending codes for special printing effects should not be confused with the "printing codes and commands" that are a part of commercially available word processing software programs. Word processing programs have built-in codes for some (but not all) special effects, such as bold type or underscoring. But Gemini can do more "tricks" than the word processing programs know to request. "Recycling" (that is, fixing) word processing programs so they are fully compatible with Gemini's capabilities requires ***"patching"*** the programs. All this is discussed in the section entitled "Word Processing Software," toward the end of this chapter.

Because of its simplicity, availability, and popularity, we will use ***BASIC*** (a computer language) to demonstrate and learn about the Gemini printer. Other languages, such as PASCAL, COBOL, FORTRAN, etc., could also be used, but to keep this training exercise short and simple, we'll stick to "BASICs."

If you are unfamiliar with BASIC, you might start by turning to the user's manual that came with your computer. There's almost certainly a chapter or two about BASIC. You might turn your attention for a few minutes to reading in your computer's manual about how to set your computer up for running a program in BASIC.

Internally, your computer uses a ***binary code*** (sometimes expressed as "zeros" and "ones") to convey information. The ASCII codes in Appendix H are ***decimal*** (in that they are based on ten numbers — 0 through 9), and ***hexadecimal*** (based on 16-number mathematical system). In our sample programs, we'll use the ASCII decimal equivalents.

In BASIC, the ***CHR$ function*** is used to convert the decimal ASCII code into your computer's binary code. Thus,

LPRINT "A" means the same as
LPRINT CHR$ (65)

The CHR$ function (pronounced "***character string***") allows you to transmit special characters and ASCII control codes that are not directly available through your computer.

## The Problem with BASIC

The "basic" problem with BASIC is similar to the problem with the ASCII code: While there are many program statements that are standard, there is no standardization of the BASIC language.

The IBM-PC, TRS-80, Atari, and many other computers (as well as Microsoft BASIC for CP/M) use the statement PRINT to send information to the screen, and the statement ***LPRINT*** to send information to the printer.

The Atari can also use a variation of the LPRINT statement, requiring (for example) that a file number (such as # 1) be assigned to the printer. Therefore, PRINT # 1 sends information to the printer, and PRINT sends information to the screen.

Other BASIC versions, such as Applesoft and CBASIC for CP/M, use PRINT to send information both to the screen and to the printer. Using the PR # 1 statement before ***PRINT*** sends information only to the printer, and using PR # 0 sends information only to the screen. Confused?

The example programs in this manual will use IBM-PC Microsoft BASIC. For those of you with other computers, we have included appendices (at the back of this manual) for the Apple II Plus, Atari, TRS-80, Commodore VIC-20 and C-64, and Osborne computers. These appendices discuss the modifications necessary to run the example programs. If your computer is not included in the appendices, look at the appendix for the system that seems most similar to your own, and look at the section in your computer's instruction manual pertaining to printers.

## GETTING "ONLINE"

You can't learn to swim in an armchair. The only way to use BASIC is to try it. So, after you've reviewed the chapter on BASIC in your

computer manual, try this exercise to "test the water." Enter this simple program:

```
LPRINT CHR$(14); "BASIC"
```

then, hit the return (sometimes called the "enter" key). That one-line program establishes communication between your computer and Gemini. Now enter this line of text:

```
LPRINT CHR$(27); "@"; "If I can do this,
                      I can do anything!"
```

After entering this line, when you hit the return key (the "enter" key on some systems), you'll be sending function codes and text to your printer. Hit the return and see what happens. If your Gemini prints out your simple program, you know that you're ready to take the plunge with the more advanced sample programs that follow.

```
If I can do this, I can do anything!
```

If, for some reason, Gemini didn't get the message and respond with action, you might do well to turn your computer off and just read over the remainder of this chapter. A wealth of information about printer capabilities is presented, even if you don't qualify as a "programmer."

If this simple test worked out well for you and your computer system, get ready because in the next example, you'll dive off the "championship" board and try a double back flip into the deep end.

## Getting Ready to Dive In

Check the position of the printer DIP switches before proceeding with the next example. For the IBM-PC, DIP switch number 2-1 on your Gemini should be "on," and the other three should be "off."

**Note:** Check with Appendices A-F for information about setting the DIP switches for various computer systems, as well as for modifications that may be required in order to run the following program.

If you are using sprocket-fed fan fold paper, roll the paper up until the perforation is even with the ribbon guide.

**Note:** Because the paper drive is engaged when the power is "on," you should only adjust the paper position when Gemini's power is "off." Moving the platen (by turning the platen knob) when the power is "on" may damage the paper-advance mechanism.

Turn your computer "on." Follow the set-up requirements, if any (such as the selection of the printer interface mode). Load BASIC and turn "on" your Gemini. The "Power," "Ready," and "Online" lamps should be lit on Gemini's control panel.

Type in the reference-program listing, Figure 5-1 making the modifications outlined in the appendices for your computer system.

**Note:** You should "proofread" each line you type in before entering it. If you want your program to look like Figure 5-2, your program listing must look exactly like the one shown in Figure 5-1 (subject, of course, to minor changes for computers other than the IBM-PC).

## Before You Run the Sample Program

The program you just typed in was a long one, but it illustrates about one-half of the features available on Gemini. Before we run the program, let's utilize Gemini as a line printer. Create a program listing by typing **LLIST.** You can use this listing to double-check that you have correctly entered the reference program (Figure 5-1). If you discover any errors, correct them. Then create another listing by entering:

```
WIDTH "LPT1:", 255
LPRINT CHR$(15)
```

and then,

```
LLIST
```

The WIDTH command told the BASIC program that your printer could print as many characters as it wants per line. The CHR$(15) was sent to tell Gemini to print in "reduced" type of 17 characters per inch (CPI). By using the reduced type, full 136-character output (Figure 5-1) can be printed on the Gemini-10X and 233 characters on the Gemini-15X.

```
10 WIDTH  "LPT1:",132
20 ESC$=CHR$(27) : LF$=CHR$(10) : FF$=CHR$(12) : CR$=CHR$(13)
30 PICA$=ESC$+"B"+CHR$(1) : ELITE$=ESC$+"B"+CHR$(2) : COMP$=ESC$+"B"+CHR$(3)
40 SUPER$=ESC$+"S"+CHR$(0) : SUB$=ESC$+"S"+CHR$(1) : CANS$=ESC$+"T"
50 ITALIC$=ESC$+CHR$(52) : STAND$=ESC$+CHR$(53)
60 WIDE$=CHR$(14) : CANWIDE$=CHR$(20) : PRSET$=ESC$+"@"
70 EMPH$=ESC$+"E" : CEMPH$=ESC$+"F" : DBLS$=ESC$+"G" : CDBLS$=ESC$+"H"
80 LPRINT PRSET$;ESC$;"R";CHR$(15);FF$
90 FOR I = 1 TO 2
100 IF I = 1 THEN LPRINT WIDE$;"       GEMINI STANDARD TYPE STYLES";CANWIDE$;LF$
110 IF I = 2 THEN LPRINT ITALICS$;WIDE$;"      GEMINI ITALIC TYPE STYLES";CANWIDE$;LF$
120 FOR N = 1 TO 40 : LPRINT CHR$(231) ; CHR$(232) ; : NEXT N : LPRINT : LPRINT
130 LPRINT"           17 cpi 12 cpi  10 cpi    8.5 cpi     6 cpi         5 cpi"
140 FOR N = 1 TO 80 : LPRINT CHR$(241) ; : NEXT N : LPRINT
150 MSG1$="  ABC" : MSG2$="Xx" : MSG3$="Yy"
160 FOR J = 1 TO 4
170 IF I = 2 THEN LPRINT ITALIC$;
180 IF J = 1 THEN LPRINT COMP$;"normal         ";
190 IF J = 2 THEN LPRINT COMP$;"double strike  ";
200 IF J = 3 THEN LPRINT COMP$;"emphasized     ";
210 IF J = 4 THEN LPRINT COMP$;"double strike  ";
220 FOR K = 1 TO 6
230 IF K = 1 THEN LPRINT COMP$;
240 IF K = 2 THEN LPRINT ELITE$;
250 IF K = 3 THEN LPRINT PICA$;
260 IF K = 4 THEN LPRINT COMP$;WIDE$;
270 IF K = 5 THEN LPRINT ELITE$;WIDE$;
280 IF K = 6 THEN LPRINT PICA$;WIDE$;
290 IF J = 2 THEN LPRINT DBLS$;
300 IF J = 3 THEN LPRINT EMPH$;
310 IF J = 4 THEN LPRINT DBLS$;EMPH$;
320 LPRINT MSG1$;SUPER$;MSG2$;CANS$;SUB$;MSG3$;CANS$;CDBLS$;
330 IF K > 3 THEN LPRINT CANWIDE$;
340 NEXT K
345 IF J=2 THEN LPRINT LF$;
350 IF J = 4 THEN LPRINT ESC$;"A";CHR$(9);LF$;COMP$;"      + ";CR$;"emphasized   ";
360 LPRINT PRSET$
370 NEXT J
380 LPRINT STRING$(5,10)
390 NEXT I
400 LPRINT FF$ : WIDTH "LPT1:",80 :END
```

*Figure 5-1. This sample program (written in BASIC for the IBM-PC, but amendable to any personal computer) is shown here in reduced or "compressed" type and will produce the "Printing Effects" shown in Figure 5-2.*

## Run the Reference Program

Because we left the printer in the 17-CPI mode at the end of the sample program listing, we need to reset Gemini before running the program.

To reset (initialize) your printer, turn the power "off," then back "on." Another way to initialize Gemini is via the software message:

LPRINT CHR$(27); "@"

A moment ago we talked about plunging into this sample exercise. The way to make that plunge is to enter:

RUN

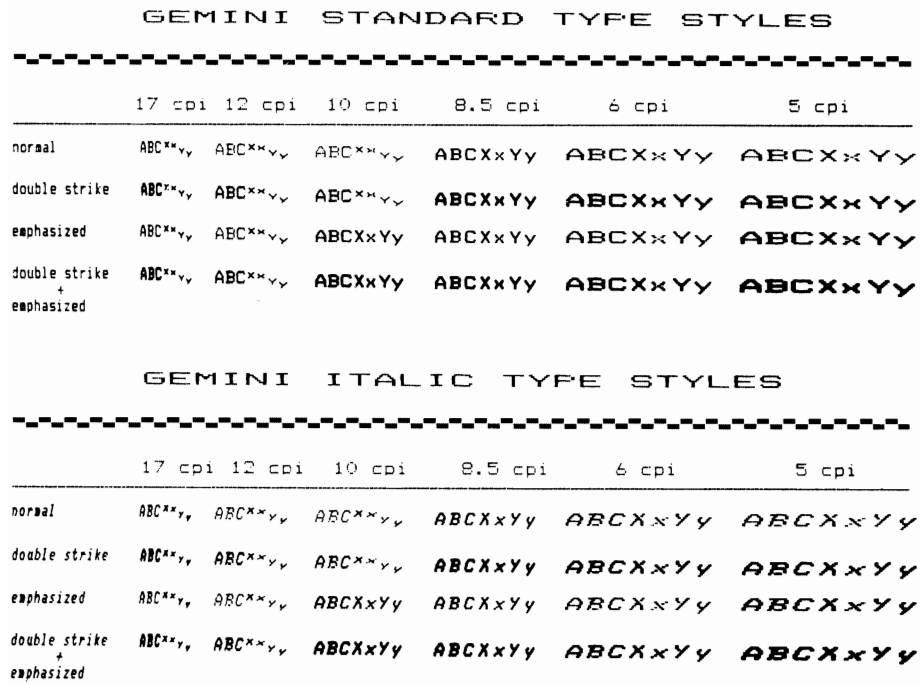

GEMINI STANDARD TYPE STYLES
17 cpi 12 cpi 10 cpi 8.5 cpi 6 cpi 5 cpi
normal
double strike
emphasized
double strike + emphasized
GEMINI ITALIC TYPE STYLES
17 cpi 12 cpi 10 cpi 8.5 cpi 6 cpi 5 cpi
normal
double strike
emphasized
double strike + emphasized

*Figure 5-2. This sample run program (shown here at two-thirds actual size) demonstrates standard and italic type styles in six sizes (measured in characters per inch) for normal, double-strike, emphasized, and double-emphasized types — each with superscripts and subscripts.*

There it is, a reference sheet that summarizes the various type styles that Gemini can produce. If your printout looks like Figure 5-2, great! Turn your printer "off," advance the paper, and tear off the tables. We will now analyze the program and printout to learn how we managed to do what we did.

### On the Other Hand...

If you made a typing error that caused the program to "belly flop," or if your printout looks strangely dissimilar to Figure 5-2, you should reset Gemini and try again. The printer may have received an unwanted function code. Or, perhaps the communication became garbled between your computer and Gemini. Whatever the problem, reset (initialize) Gemini and try again.

## ANALYZING THE PROGRAM

In this section, we'll analyze the program you entered, using Figure 5-1 as your reference. Before we start, you might want to **SAVE** the program on disk or tape to avoid having to retype it if you ever want to use it again.

Let's begin by considering line 10, shown in Figure 5-1. This line tells the BASIC interpreter that the printer will accept 132 characters per line. This prevents BASIC from tossing in extra carriage returns and line feeds where we don't want them.

Lines 20 through 70 are used to set some printer controls to more easily remembered codes. The escape code (which is ASCII decimal code 27) is set to **ESC$**. The command that tells the printer to advance one line (ASCII code 10) is set to **LF$**. The variable **FF$** is set to ASCII code 12, a command to Gemini that the printer should advance to the top of the next page. The default value, set by power "on" or receipt of the software reset, is 66 lines per page.

Line 30 sets the printer command to change type size. The format of this command is ESC$ + "B" + n, where n can be 1, 2, or 3. If n is 1, the type size is 10 characters per inch (CPI). This is the type size of a standard typewriter's print, called "pica." Pica is the type size you select when Gemini is initialized, until and unless you ask for different type size. If n is 2, the type size is 12 CPI, the same size as "elite" typewriter print. If n is 3, the type size is 17 CPI, the size of the compressed type we used for the listing in Figure 5-1.

Wait a minute. Didn't we use the command code CHR$(15) to shift into compressed type? That's right; there are two ways of shifting into **compressed** type: — the single-character command; CHR$(15), and the three-character command; ESC$ + "B" + 3. [Actually, that's CHR$(27); CHR$(66); and CHR$(3), if you check the ASCII code table in Appendix I.] The command CHR$(18) can be used to shift the printer from compressed mode back to normal pica-size print.

Line 40 sets some variables that allow shifting into (and out of) the **superscript** and **subscript** mode. Superscripts are characters that are printed above the regular line of type, such as the "2" in $E = MC^2$. Subscripts are below the line, such as the "2" in $H_2O$. On some printers, the "supers" and "subs" are produced by rolling the paper up and down a half-line, printing the superscript or subscript, then rolling the paper back to its original position.

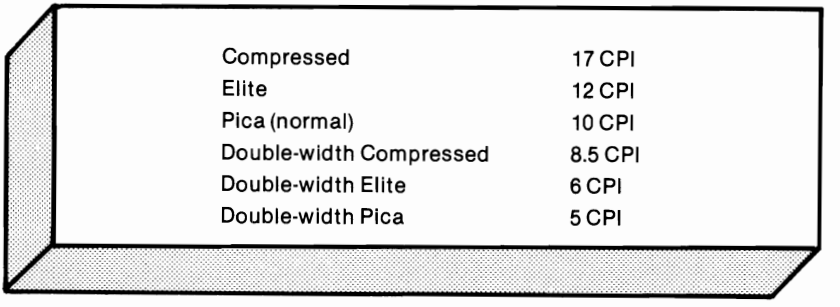
Gemini prints "supers" and "subs" using special characters that are small enough to convey the scripting in the space normally utilized by the ascenders 'such as the top of a "t") and descenders (the lower part of the letter "g"). The scripting characters are printed "high" or "low" on the line to produce a professional-looking scripting effect — without having to roll the paper up and down by hand.

ESC$ + "S" + CHR$(0) causes the printer to operate in its superscript mode, and ESC$ + "S" + CHR$(1) initiates the subscript. Either mode is cancelled by ESC$ + "T" command — actually, CHR$(27); CHR$(84).

Line 50 sets the command to produce **italic type.** CHR$(27); CHR$(52) causes Gemini to print in full italic all text received subsequent to the command, until the ESC$ + CHR$(53) command [or printer reset = CHR$(27); "@"] is received.

Line 60 sets the command to **double the width** of the characters that are being printed. CHR$(14) sets the double-width print mode, and CHR$(20) cancels the double-width mode. ESC$ + "W" + 1 [which is the equivalent of CHR$(27); CHR$(87); CHR$(1) command] also sets the double-width mode; in this case, ESC$ + "W" + 0 cancels the double-width mode. You might note that with the three type sizes, plus the double-width mode, we have six different character sizes from which to select.

On the end of line 60 is the printer reset command we call "PRSET" in this program.



| | |
|---|---|
| Compressed | 17 CPI |
| Elite | 12 CPI |
| Pica (normal) | 10 CPI |
| Double-width Compressed | 8.5 CPI |
| Double-width Elite | 6 CPI |
| Double-width Pica | 5 CPI |

Line 70, the last line of our definitions, contains functional commands to control the way the type is printed. In emphasized printing, Gemini prints the characters once, then goes back and reprints them (with the dots offset by one-half dot). This is explained in the next chapter, but for now just remember that both emphasized and double-strike involves printing each character twice.

In line 70, ESC$ + "E" puts the printer in the ***emphasized*** mode, and ESC$ + "F" cancels the emphasized printing. ESC$ + "G" puts the printer in ***double-strike*** mode, and ESC$ + "H" cancels that mode.

Both ***emphasized and double-strike*** can be invoked at the same time to produce maximum-density printing. Some restrictions apply to the use of emphasized printing and some of the other print commands. We'll go over these restrictions when we examine in detail the tables printed by this program.

Line 80 is the first line of the program that sends "text" data to the printer. What information are we sending Gemini? First, we send the character string PRSET$, telling Gemini to reset according to the command CHR$(27); "@." Even though we didn't need to reset Gemini after printing the listing in compressed type, *it's a good programming practice to make sure your printer is in its initialized state at the beginning of your program.*

The ESC$; "R"; CHR$(15) sequence in line 80 tells Gemini to advance the paper 15 lines after the form-feed command is received. The default for the beginning line (or "***header***") is the first line on the page. The FF$ variable sends a form feed command to Gemini. This line is used to cause Gemini to skip one page of paper (that's 66 lines with conventional fanfold sprocket-fed

paper), then skip down 16 lines from the top of the next page. So where, you might ask, did the extra line come from? BASIC inserts a carriage return (***CR***) and line feed (***LF***) at the end of every LPRINT statement, unless the last character in LPRINT is a semicolon (;).

Line 90 starts the major loop of the program, which causes the normal and italic tables to be printed. If the variable "I" is equal to the number 1, then the message on line 100 is printed. If "I" equals 2, the message on line 110 is printed. Let's look at line 100:

```
LPRINT WIDE$; "GEMINI STANDARD
TYPE STYLES"; CANWIDE$; LF$
```

You might ask: What does WIDE$ cause to be printed? The answer is not "what," but "how." WIDE$ is set (on line 60) to CHR$(14), the command for the double-width printing mode. Look at Figure 5-2 to verify that.

The CANWIDE$ string cancels the double-width function (also set on line 60). *The **semicolon** between character strings causes the strings to be printed without any spaces between them.* The LF$ string causes a line feed; the LPRINT statement produces another carriage return and a line feed because it does not end with a semicolon.

Line 110 causes a similar message to be printed, but this time we did not invoke the italic print style by sending the ITALIC$ command string to Gemini at the beginning of LPRINT statement.

Line 120 looks interesting, because it serves an interesting purpose. Line 120 is a loop from 1 to 40 that sends two characters to Gemini for each cycle of the loop. What, you might ask are the two characters: CHR$(231) and CHR$(232)? To find out, turn to Appendices I or H. ASCII code numbers 231 and 232 appear to be what, in a gymnast's parlance might be called "uneven parallel bars." Number 231 is the upper bar, and number 232 is the lower bar. These two characters are part of **Gemini's "6 x 6" graphics set** (as explained further in Chapter 7).

The two LPRINT statements at the end of line 120 cause two line feeds after the loop is completed. Did you notice the *semicolon* at the LPRINT, inside the loop? Without that ";" a carriage return would be generated after each printing of these two characters. With the semicolon, we get the alternating row of short bars that serve as decoration beneath the table titles in Figure 5-2.

Line 130 is straightforward; it prints the column headings for the table. Because the printer has been set to produce these headings in I = 1 and in I = 2, the headings appear in normal and

italic type, respectively, in Figure 5-2. You might note that the decorative graphic characters are unaffected by the italic control, and the italic mode is resumed after 6 x 6 characters are printed.

Line 140 is similar to line 120, and serves as one way of drawing a line across a page.

Line 150 sets three character strings in preparation for the printing that follows. The first string, "ABC," is printed in normal script. The next string, "Xx", is printed in superscript. The third string, "Yy," is printed in subscript. If you prefer other characters (for example, your name) to be printed in another version of Figure 5-2, change these strings and rerun the program.

Line 160 starts the loop-over printing modes. Line 170 puts the printer into italic type when I = 2. This is required because we will use the printer-reset to clear the print modes at the end of this loop. The printer-reset causes Gemini to go back to its initial power "on" state.

Lines 180 through 210 print row headings for the tables, on the left side of the lines illustrating the print modes in Figure 5-2. For $J = 1$, the print mode is normal. $J = 2$ produces the double-strike effect. The print is emphasized as a result of $J = 3$. With $J = 4$, Gemini produces a combination of double-strike plus emphasized type. You might also note that we set the row headings in compressed type to conserve space so that the tables would fit on 10-inch-wide paper.

Line 220 starts the inner loop, which prints each line of the table. Lines 230 through 280 set the six different print sizes. If $K = 1$, line 230 sends the COMP$ command to Gemini, which causes a shift into the compressed print mode. If $K = 2$, then line 240 sends the ELITE$ command to the printer. If $K = 3$, line 250 sends the PICA$ command, thus directing Gemini to print in normal (10 CPI) type size. The question might arise: Could we use the PRSET$ mode instead? If we did, it would be necessary to reset the italic mode before printing the table characters when I = 2. When $K = 4$, line 260 sends both the COMP$ and the WIDE$ commands, causing the printer to shift into 8.5 CPI.

Lines 270 and 280 cause Gemini to print 6 CPI when $K = 5$, and 5 CPI when $K = 6$.

Line 290 invokes the double-strike mode when $J = 2$ by sending the character string DBL$ to Gemini. You may recall the DBL$ is equal to ESC$ + "G" or, equivalently, CHR$(27); CHR$(71).

Line 300 sends the emphasized mode command to the printer

by means of a character string EMPH$. When $J = 4$, line 310 shifts Gemini into the double-strike and emphasized mode by sending both the DBL$ and EMPH$ commands.

Line 320 is the *heart of the program.* This line prints out the entire table. The rest of the program prints the titles and headings, or sets the print size and style. In line 320, the character string MSG1$ (set to "ABC" by line 150) is printed in whatever mode the printer is in as a result of preceding the transmission of the SUPER$ command string (remember line 40?) and the character string MSG2$ is printed. The superscript mode is cancelled by sending the CANS$ string [CHR$(27); CHR$(84), or by ESC$ "T"].

The subscript mode is invoked by sending the SUB$ command string. The third character string in line 320, MSG3$ (which is "Yy"), is then printed, only to be cancelled by the CAN$ command again. Finally, the double-strike print mode is cancelled.

Why did we have to cancel the double-strike mode? Where did we invoke it? In order to make the *half-height* superscript and subscript characters more legible, they are automatically printed using the double-strike print mode. Cancelling the superscript or subscript does not automatically cancel the double-strike mode.

You might notice that line 320 is used six times to print one line. Line 330 is used to cancel the double-width command when $K$ is greater than 3. This is done because the double-width command is sent when $K$ is equal to 4, 5, or 6, by lines 260, 270, and 280.

Line 340 is the bottom of the inner $K$ loop. Line 350 is responsible for printing the words "+ emphasized" after the last line of each table. You might want to direct your attention to Chapter 9, or to Appendix M, to learn the purpose of the ESC "A" *n* functional control code. We used this control code to ***reset the spacing between lines.***

When Gemini is initialized, the spacing is set to one-sixth of an inch, when DIP switch 1-5 is "on." (This is the standard "single-space" of a conventional "pica" typewriter.) With Gemini, this 1/6-inch spacing allows a "half-character" of spacing between lines of text.

The ESC$; "A"; *n* function code allows us to vary the spacing between lines. When the spacing control codes are sent to the printer, the spacing for the line feed is set to *n*/72 inch. Thus, if *n* is set to be 12, the spacing in 12/72, which is 1/6 of an inch. If *n* is less than 12, the spacing between lines will be less than the normal

default spacing. If *n* is greater than 12, the spacing will be larger than normal.

In line 350, we set *n* to equal 9, with the result that the line spacing is changed to 9/72 or ⅛ inch. A line feed is set to move down (or, to be more correct, "advance the paper") ⅛ inch, then return the print head to the first print column (it's "***home***" position). The printer is shifted into the compressed mode because of COMP$, the "+" is printed, followed by another carriage return and line feed. The word "emphasized" is then printed, and line 350 has done its job.

Line 360 clears all the previous print function codes and restores Gemini to its initialized values. Line 370 is the bottom of the *J* loop.

Line 380 is one way of ***advancing the paper*** by multiple line feeds. The STRING$(5,10) function is used to return a string of length *n*, whose characters all have the ASCII code of the first character in A$. Because the line-feed command is a single character, this function allows us to skip multiple lines. Vertical tabs and other line-spacing controls are discussed further in Chapter 9.

Line 390 is the bottom of the outer loop. Line 400 finishes up the program by sending a form-feed command to Gemini, resetting the "line printer" to an 80-character width. With that, line 400 ends the program.

Now that we have considered the *causes,* let's look at the *effects.* In the following section, we'll interpret Figure 5-2, amended to show the 88 type styles and sizes that Gemini can provide for practical applications.

## WHAT WE'VE LEARNED GEMINI CAN DO

So far in this chapter, we've learned that Gemini can do more than the most popular word processing software can handle. To enjoy the full range of type styles and sizes, you need to create computer programs in BASIC (or other computer languages) to send functional commands to Gemini. Or, as described in the following section, you can "patch" your word processing software so it's smart enought to manage Gemini properly.

Figure 5-3 is an amended version of Figure 5-2 that shows the ***88 variations of type sizes and styles*** for the alphanumeric

characters. ("Alphanumeric" means all the characters that are the letters in the *alphabet* and numerical symbols for numbers.) The difference between Figures 5-2 and 5-3 is that the latter has had some of the superscripts and subscripts eliminated.

The "supers" and "subs" are designed for normal and double-strike type in 10, 12, and 17 CPI. Superscripts and subscripts were not intended for use in the emphasized mode. When you double the width of the characters (5, 6, and 8.5 CPI), the "scripts" are distorted to the point that they are garbled. They weren't designed for double-width printing, and aren't needed.

By examining Figure 5-3, you can see at a glance that Gemini can print in **six type sizes,** between 5 and 17 characters per inch. Gemini prints in roman type (the upper table) and italic type (the lower table). In addition, Gemini produces quality superscripting in 20 modes, as well as quality subscripting in an equal number. If you add up all these typeface options, you'll count as many Gemini text options as there are keys on a piano.

Figure 5-3 also demonstrates a hint of Gemini's 6 x 6 graphics capability. If you look in Appendix K at ASCII code numbers 224 through 255, you'll see an enlargement of the full set of 6 x 6 graphics. The straight line that separates the CPI column headings from the body of the tables in Figure 5-3 was produced by repeating ASCII code number 241 across the page. The alternating sequence of ASCII code numbers 231 and 232 produced the graphic embellishment that appears under the "standard" and "italic" table titles. In Chapter 7, you'll learn more about what you can do with the 6 x 6 graphic character set.

Although the "ABC's" in Figure 5-3 vary in intensity, slant, and width, they are all the same height. That's because all Gemini's alphanumeric characters are printed using a **fixed-height** matrix.

The matrix is represented by the numerals that appear on the first page of each chapter in this manual. A reduced version of that chapter-identifier appears at the top of this page. You can see that the matrix (or grid) is nine rows high and five rows wide (with some dots on the lines between columns). The compressed and double-width modes print the same character matrix, but with the width of the horizontal columns shortened or expanded, respectively.

If you are observant, you might notice that the "ABC's" across each row in Figure 5-3 appear to slope slightly downward from left to right. In your everyday printing, you won't get this effect; your lines of text will be perfectly straight. The reason for

the sloping in this sample program is the frequent use of superscripts and subscripts.

When you print "supers" and "subs," Gemini executes the "scripts" in the double-strike mode. Double-striking involves printing the characters once, then reprinting them 1/144th inch lower. One one-hundred-and-forty-fourth of an inch isn't even detectable (unless you have the eyes of a hawk).

In most printing situations, you won't use superscripts and subscripts more than once or twice a line (if at all). Thus, printing the remainder of a line 1/144th or 2/144th of an inch lower would not be noticeable. In our sample program (Figure 5-2), there were 12 "supers" and "subs" per line — giving a total of 12/144ths (or

GEMINI STANDARD TYPE STYLES

| | 17 cpi | 12 cpi | 10 cpi | 8.5 cpi | 6 cpi | 5 cpi |
|---|---|---|---|---|---|---|
| normal | ABC$^{x_x}_{y_y}$ | ABC$^{x_x}_{y_y}$ | ABC$^{x_x}_{y_y}$ | ABC | ABC | ABC |
| double strike | ABC$^{x_x}_{y_y}$ | ABC$^{x_x}_{y_y}$ | ABC$^{x_x}_{y_y}$ | ABC | ABC | ABC |
| emphasized | ABC$^{x_x}_{y_y}$ | ABC$^{x_x}_{y_y}$ | ABC | ABC | ABC | ABC |
| double strike + emphasized | ABC$^{x_x}_{y_y}$ | ABC$^{x_x}_{y_y}$ | ABC | ABC | ABC | ABC |

GEMINI ITALIC TYPE STYLES

| | 17 cpi | 12 cpi | 10 cpi | 8.5 cpi | 6 cpi | 5 cpi |
|---|---|---|---|---|---|---|
| *normal* | *ABC$^{x_x}_{y_y}$* | *ABC$^{x_x}_{y_y}$* | *ABC$^{x_x}_{y_y}$* | *ABC* | *ABC* | *ABC* |
| *double strike* | *ABC$^{x_x}_{y_y}$* | *ABC$^{x_x}_{y_y}$* | *ABC$^{x_x}_{y_y}$* | *ABC* | *ABC* | *ABC* |
| *emphasized* | *ABC$^{x_x}_{y_y}$* | *ABC$^{x_x}_{y_y}$* | *ABC* | *ABC* | *ABC* | *ABC* |
| *double strike + emphasized* | *ABC$^{x_x}_{y_y}$* | *ABC$^{x_x}_{y_y}$* | *ABC* | *ABC* | *ABC* | *ABC* |

*Figure 5-3. The sample program's results, amended to show the 88 viable options for type sizes and styles using Gemini.*

1/12th) of an inch difference between the beginning and end of the line.

By examining the lower table in Figure 5-3, you'll see that the same options and parameters apply to italic type as to standard type.

## WORD PROCESSING SOFTWARE

Word processing is the family of programs that package text in forms understandable by your microcomputer, its display monitor, and various printers. Popular word processing programs include Wordstar, the Electric Pencil, Easywriter, Word Handler, Perfect Writer, etc. (Note: Each of these programs is the copyrighted property and registered trademark of the companies that own them. They are mentioned here only as examples of commercially available word processing programs.)

Most of the popular word processing programs were written a few years ago, when printers were simpler devices, capable of performing far fewer options than are available with Gemini.

For example, most current word processing programs can send function codes for printing "bold" type (double-striking). To accomplish this, the word processing software sends multiple instructions (initiated by keyboard commands such as "control-B" or "control-E"). When older printers receive these instructions, they print a word, then get the signal for their printing mechanism to back up, then respond to another signal that tells the "dumb" printer to print the word again. With Gemini, the function code ESC "E" tells the smart printer to automatically **double-strike** using the LPRINT command conveyed in BASIC.

Don't be discouraged if you bought a word processing package and it seems inadequate to utilize Gemini's full capabilities. Gemini can do just about anything your word processing software can tell it to do, and then some. There are four solutions to the problem.

### The BASIC Solution

One solution to the limitations of word processing software is not to use it when you want to make full use of Gemini's potential performance. You can learn BASIC and put it to work.

Even without "mastering" BASIC, you'll be able to work wonders with Gemini, as you've already learned in this chapter.

## User-Patching Your Software Program

"Patching" is the term used to describe making changes in an existing word processing program so that it can make fuller use of Gemini's capabilities. To accomplish patching, refer to your word processing software user's manual. The manual should have a listing that demonstrates "user-patchable" changes you can make in the program.

A word of **warning:** You need to be computer-wise to successfully patch an existing word processing program. Before you start, it's probably a good idea to make a copy of your word processing program and attempt patching the copy. That way, if you make an error, you're only out the time you invested and haven't hurt the "original."

## Software Patching Services

Another solution to the problem of software/printer incompatibility is to turn to one of the "software recycling" services that advertise in hobbyist computing publications.

For a modest sum, they will "recycle" your software by performing the patching operation for you. Many of these services invite you to mail them your word processing software (and, of course, a check), and they'll send you back a "tailored-to-fit" program on your same diskette(s).

## The Creative Solution

If you made it through the program listed in Figure 5-1, you should be able to create a simple program that serves as

"Your Own Word Processing Software System."

(The title's a bit long; you can always "word process" it to something short and snappy.) Even if you didn't successfully execute the sample program, don't be afraid of trying this task. Remember: You may never know if you can do something unless you try.

```
10 CLS:KEY OFF:DIM A$(24,80)
20 CLS:X=1:Y=1
30 GOSUB 150:LOCATE X,Y:PRINT "@";
40 B$=INKEY$:IF B$="" THEN 40
50 IF B$=CHR$(5) THEN CLS:END
60 IF B$=CHR$(18) THEN CLS:GOSUB 200:GOTO 20
70 IF B$=CHR$(16) THEN GOTO 220
80 IF B$=CHR$(13) THEN A$(X,Y)=B$:LOCATE X,Y:PRINT " ":X=X+1:Y=1:LOCATE X,Y:PRINT
   "@";:GOTO 40
90 IF B$=CHR$(4) THEN GOSUB 130:GOTO 40
100 LOCATE X,Y:PRINT B$;:A$(X,Y)=B$:Y=Y+1
110 IF Y=80 THEN A$(X,Y)=CHR$(13):BEEP:X=X+1:Y=1:GOTO 40
120 LOCATE X,Y:PRINT "@";:GOTO 40
130 LOCATE X,Y:PRINT " ";:Y=Y-1:IF Y=0 THEN Y=1:BEEP:
135 A$(X,Y)=""
140 LOCATE X,Y:PRINT "@";:RETURN
150 LOCATE 25,2:COLOR 0,7:PRINT " ^P) printer ";:
160 LOCATE 25,17:COLOR 0,7:PRINT " ^D) delete ";:COLOR 7,0
170 LOCATE 25,34:COLOR 0,7:PRINT " ^R) restart    ";:COLOR 7,0
180 LOCATE 25,51:COLOR 0,7:PRINT " ^E) exit      ";:COLOR 7,0
190 RETURN
200 LOCATE 10,30:PRINT "please stand by";:FOR I=1 TO X:FOR J=1 TO 80:A$(I,J)="":
    NEXT:NEXT:RETURN
220 LPRINT CHR$(27);"@"
230 CLS:LOCATE 25,2:COLOR 0,7:PRINT " C)ommand ";:LOCATE 25,16
240 PRINT " P)rint ";:LOCATE 25,27:PRINT " R)eturn ";:COLOR 7,0
250 G$=INKEY$:IF G$="" THEN 250
260 IF G$="P" THEN GOSUB 310
270 IF G$="C" THEN GOSUB 320:GOTO 230
280 IF G$="R" THEN GOSUB 300:GOTO 30
290 GOTO 250
300 CLS:FOR I=1 TO X:FOR J=1 TO 80:LOCATE I,J:PRINT A$(I,J);:NEXT J:NEXT I:RETURN
310 FOR I=1 TO X:FOR J=1 TO 80:LPRINT A$(I,J);:NEXT J:NEXT I:RETURN
320 LOCATE 25,2:PRINT STRING$(50,32);
330 LOCATE 25,2:PRINT STRING$(50,32);
340 LOCATE 25,2:COLOR 0,7:PRINT " E) Esc ";:LOCATE 25,11
350 PRINT " C) CHR$ ";:LOCATE 25,21:PRINT " Q) end command ";:COLOR 7,0:ST$=""
360 G$=INKEY$:IF G$="" THEN 360
370 IF G$="E" THEN GOSUB 420:GOTO 400
380 IF G$="C" THEN GOSUB 530:GOTO 400
390 IF G$="Q" THEN GOTO 410
400 GOTO 360
410 RETURN
```

```
420 I=0:LOCATE 23,4:PRINT "Esc ";
430 G$=INKEY$:IF G$="" THEN 430
440 IF G$=CHR$(13) THEN 480
450 PRINT G$;
460 PRINT " ";
470 S$(I)=G$:I=I+1:GOTO 430
480 IF I>2 THEN FOR J=1 TO I:D$=D$+S$(J):NEXT J:ST$=CHR$(27)+S$(0)
    +CHR$(VAL(D$)):D$=""
490 IF I=2 THEN ST$=CHR$(27)+S$(0)+CHR$(VAL(S$(1)))
500 IF I=1 THEN ST$=CHR$(27)+S$(0)
510 LPRINT ST$;ST$="":FOR J=1 TO I:S$(J)="":NEXT J:I=0
520 LOCATE 23,4:PRINT "                                    "; :RETURN
530 LOCATE 23,2:PRINT "CHR$ ";:ST$=""
540 G$=INKEY$:IF G$="" THEN 540
550 IF G$=CHR$(13) THEN 570
560 PRINT G$;:ST$=ST$+G$:GOTO 540
570 LOCATE 23,2:PRINT "                          ":LPRINT CHR$(VAL(ST$));:ST$="":RETURN
```

*Figure 5-4. A simple program, written in BASIC, that serves as a rudimentary word processing program.*

In Figure 5-4, a very simple word processing software program is presented, written in BASIC for the IBM-PC. This program leaves some room for improvement, but after all — it's free. You might apply it to writing short letters or other simple documents. The display on your CRT monitor may not be anything to write home about, but at least it gives you the capability of writing home! You may also find the use of the control codes is somewhat awkward, but this program has its own menus which will make it easier for you to remember the commands. The program provides three different menus which display your options and prompt you to select specific commands to guide you in your word processing. After you have typed in the program and entered RUN followed by a carriage return, the first menu will appear at the bottom of the screen and the cursor should be at the top left corner.

Now you can type in your text and correct any errors you may make. Because of the simplicity of this program, it does have some limitations, which will vary for the different computers. If your computer is an IBM-PC, you can type in 23 lines at up to 80 characters per line. Some other computers may not have the capacity to accept this many characters, limiting the amount of text you can type. This program is also restricted in that errors can only be corrected on the present line. Once you have hit <return> you cannot go back to the previous line to revise it.

After typing your text, you may go either to the second menu, which will permit you to print your text, or to the third menu where you may send Escape or control codes to the printer for special characters or effects. It is not necessary to type the entire ASCII codes. Instead, you have the option of selecting "ESC" or "CHR$". For example, if you wish to send CHR$(27); "W"; CHR$(1); to the printer for printing of wide characters, you simply need to select the "ESC" option and type "W1" and then hit <CR>, and the code will be sent to the printer. After sending the codes, you may go back to the 2nd menu and print your text again.

The following is a listing of the commands:

CONTROL COMMANDS FOR THE "DO-IT-YOURSELF" WORD PROCESSING PROGRAM (SHOWN IN FIGURE 5-4)

1st MENU

^P = gets you to the 2nd command level
^D = deletes the character to the left of the cursor
^R = restart: clears the screen and lets you type a new text
^E = exits the program

2nd MENU

P = prints your text
C = gets you to the 3rd command level
R = returns you to the 1st print level

3rd MENU

E(ESC) = lets you send ESC code with an abbreviated command
C(CHR$) = lets you send a control code with an abbreviated command
R = returns you to the 2nd level

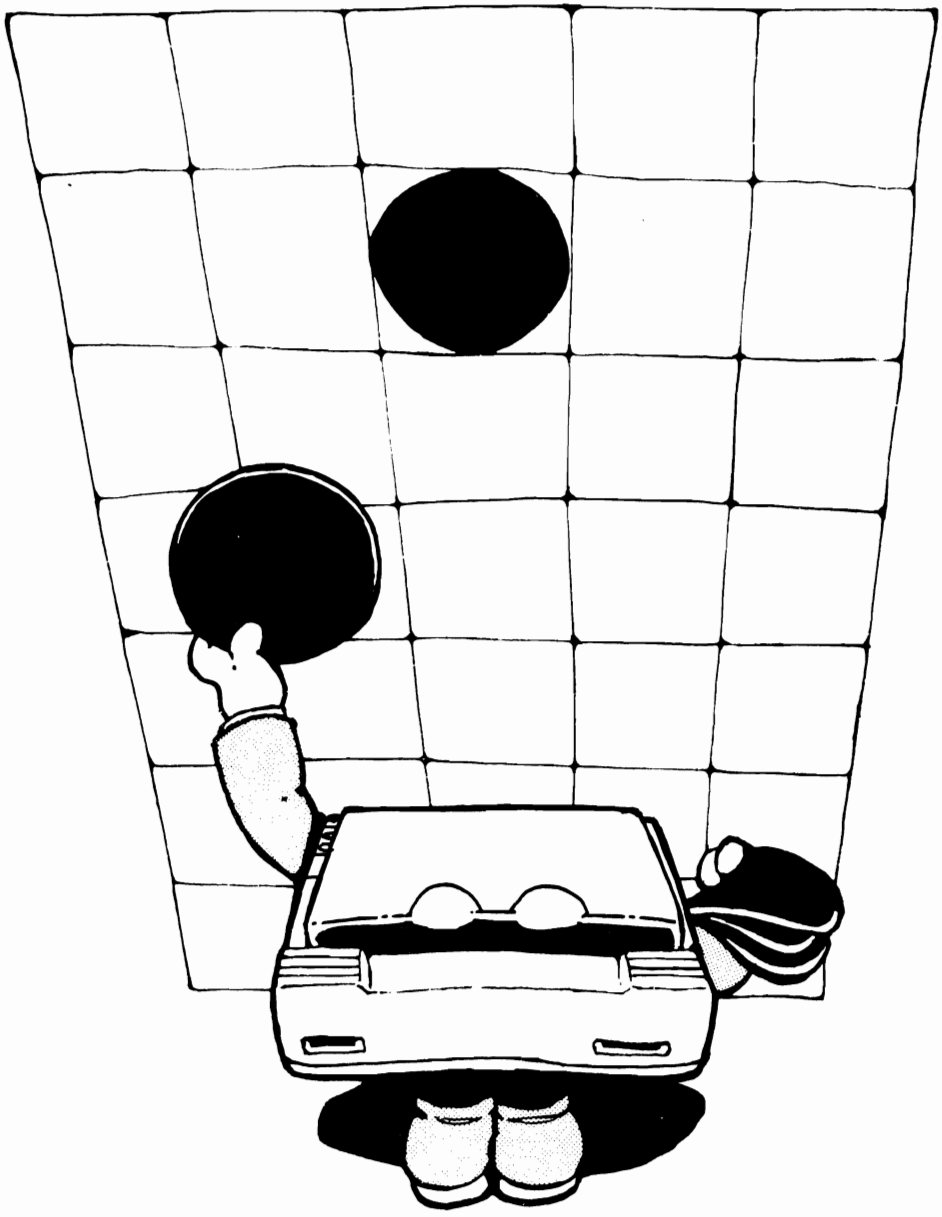
These commands may vary if you are not using an IBM-PC, but the effects will be the same. For further explanation of the program variations, please refer to the computer-specific appendices at the back of this manual.

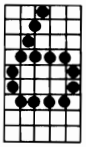
## WHERE DO WE GO FROM HERE?

If you want to get to work, you may take your leave at this point. You've gotten a good idea of what Gemini is all about — how the parts fit together, how to add paper and ribbon, the Self-Test, interfacing with your computer, and various printing capabilities. That should be enough to get you started.

To paraphrase an old saying: "A journey of a thousand pages begins with a single character." *Bon Voyage!*

If you have the luxury of enough time to keep reading, the next five chapters and accompanying appendices will provide you with the means to be able to travel first class. If knowledge is power, then you'll be a powerful Gemini user by the end of this manual.





# THE FUNDAMENTALS OF DOT MATRIX PRINTING

Understanding dot-matrix printing and Gemini's graphics begins with knowing some details about the print head and the software that controls it. This chapter will provide you with those details. The print head, print matrix, and vertical spacing are described in understandable terms. The second half of this chapter will introduce you to dot graphics: the control codes for firing the "pins," and a sample graphics program that will put a smile on your paper, and perhaps on your face.

## THE PRINT HEAD

The print head contains nine thin wires, or pins, positioned one above the other. Figure 6-1 illustrates the print head, from the paper's point of view, looking straight on. Each wire is connected to a solenoid "gun," an electromagnet that "fires" when told to do so by the "computer" inside Gemini. Figure 6-2 presents a schematic of the print-head mechanism, profiling the solenoid, spring, and dot-wire pin for one of the guns.

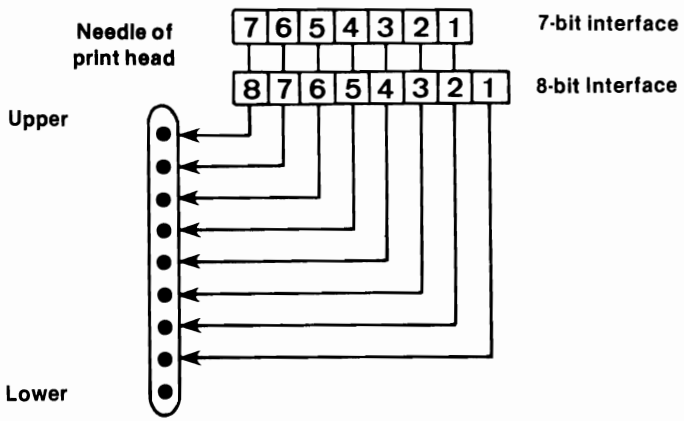



*Figure 6-1. The bit-image column-scan capability is the result of nine pins, stacked in a column, each "fired" independently in response to signals from the 7-bit or 8-bit interface from your computer.*

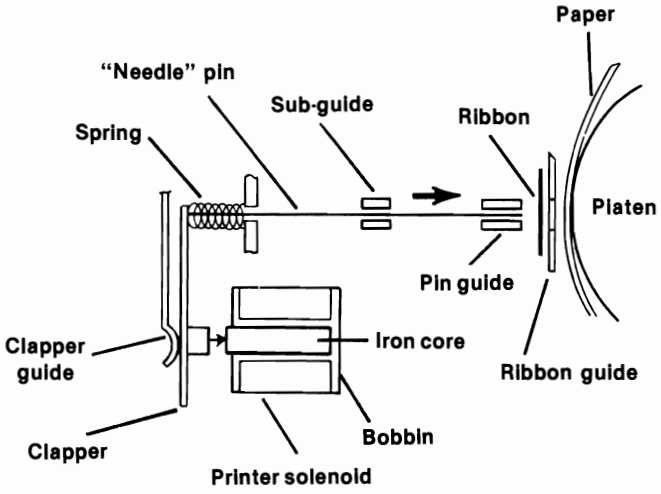



*Figure 6-2. This side-view schematic of the print-head mechanism shows how electrical signals to the solenoid cause the clapper to "fire" the needle-like pin that impacts the ribbon/paper/platen. After the solenoid releases the clapper, a spring returns the pin to its original position.*

Gemini's built-in "computer" reads the ASCII code instructions sent by your computer and directs the print head to discharge the correct pins in the proper order. The result of this "hiring" and "firing" is the printed characters produced by Gemini onto your paper.

Consider, if you will, the process of printing an uppercase letter "H" (see Figure 6-3). Because the print head is bidirectional (can print right to left, and left to right), Gemini can start the "H" from either side.

In this example, the top seven pins fire simultaneously to print one side of the letter. Then, the print head moves across the paper, and pin number four fires three times to create the horizontal line. Finally, the top seven pins fire simultaneously again to complete the other vertical line. Aside from the amazing fact that the entire printing operation just described can take place in approximately one-hundred and twentyth of a second, all this activity is caused by a single ASCII code signal.

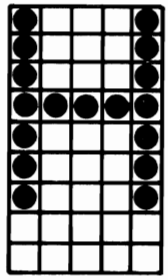


*Figure 6-3. The nine rows of the matrix correspond to the nine pins on the print head. For this uppercase letter, only the top seven pins fire — all at once for each vertical line in this letter, and a single pin (firing repeatedly as the print head moves sideways) for the horizontal line.*

This printing technology is called impact, dot matrix printing. Dots are printed according to a predesigned matrix or grid system. Each letter, numeral, punctuation mark, and other special character is formed by the arrangement of 4 to 36 dots.

The firing patterns for each character are pre-programmed in Gemini and stored in "firmware" (ROM's). By sending the appropriate control codes to Gemini, we manipulate both the number of pins fired and the order in which they are fired. If you know which control code to send from your computer's keyboard, through the computer to the printer, you can even create new "characters" by controlling the firing pins.

## THE PRINT MATRIX

Let's turn to Appendix K and look briefly at the dot patterns of the various characters. All these characters, during the "normal" printing mode (standard and italic), are constructed within a 5-dot-wide by 9-dot-high matrix of "boxes." All the dots are positioned vertically within one of the "boxes" in each column in the matrix.

Horizontally, the dots can be placed within any or all of the five boxes across the row or on the lines separating the boxes. If you think of the five boxes in a row as the "white" keys on a piano and the lines between them as four "black" keys, you can "play" a dot in any of the nine dot positions across a row. Thus, even

though there are only five boxes across, you have "9x9" printing positions.

The uppercase letters are formed by the top seven pins. Some of the lowercase letters ("g," "j," "p," "q," and "y") extend down to the eighth and ninth pins. This style of character generation, called "true descenders," makes these letters much more readable.

The "underlining" character (ASCII code 95) is created by firing only the ninth pin. The ESC CHR$(48) control code can also be used for underlining.

The block graphics characters (ASCII code numbers 224 through 255) are configured on a "6 x 6" matrix. We'll discuss these characters a little later. A simple example of block graphics is shown in Figure 6-4: "W" "y" for the "Cowboy" state. (The BASIC program for producing Figure 6-4 is presented and discussed in Chapter 7.)

*Figure 6-4. Using block graphics and a simple program written in BASIC, you can create very large type, as shown here.*

Although the matrices are the same width, the dot configurations of the different characters in the alphanumeric and special symbol sets are different in their widths. The narrowest letter is the lowercase "i" (ASCII code 105); the smallest character is the four-dot period (ASCII code 46). The widest characters in the alphanumeric set are the uppercase letters, such as "M" (ASCII code 77).

When Gemini prints these characters, for example, during the first lines of the Self-Test, an extra "blank" column (a sixth column) is added to each character to provide spacing between them.

When you shift into the elite (12 CPI) or compressed mode (17 CPI), the characters are the same height as in the normal mode, but vary in width. With elite and compressed printing, the "columns" of the matrix become narrower.

With emphasized type, the matrix is 18 by 9 because each column is fired twice, horizontally offset by one-half dot on the second firing. Double-strike printing is accomplished by an 18 by 18 matrix, with the second printing vertically and horizontally offset by one-half dot. Emphasized plus double-strike is a 36 by 27 matrix, with the additional printings offset both horizontally and vertically.

In the double-width ("expanded") mode, a column is printed, the print head moves over and repeats the same column configuration, then the print head moves over to twice-print the next column.

Superscripts and subscripts follow the same matrix pattern as the standard characters, but the print size of the matrix is one-half as high.

## VERTICAL SPACING

On most conventional typewriters, you have a choice of single, space-and-a-half, or double spacing. Your Gemini can print lines that are set to one-sixth inch, or one-eighth, or whatever spacing you want in increments as small as 1/144th of an inch!

When you "initialize" your printer, Gemini is automatically set to a vertical spacing of one-sixth inch if DIP Switch 1-5 is "on" or one-eighth inch if DIP Switch 1-5 is "off." One-sixth inch spacing, which delivers six lines per inch, is the same as a standard "pica" typewriter, and is fine for text printing.

For block graphics (each of which is up to six dots tall), different spacing is required (or else your graphic "blocks" won't fit together tightly).

Anticipating that you would want to be able to adjust the vertical spacing, Gemini provides you with the following five ways to make adjustments:

| **Software Code** | **Height (Inches)** | **Vertical Dots** |
|---|---|---|
| ESC "2" | 1/6 | 12 |
| ESC "0" | 1/8 | 9 |
| ESC "1" | 7/72 | 7 |
| ESC "A"*n* | *n*/72 | *n* |
| ESC "3" *n* | *n*/144 | *n*/2 |

As if that weren't enough, there's ESC "J" *n,* which does the same work as ESC "3" *n* but doesn't change the value for successive line feeds.

Because each "dot" is 1/72-inch in diameter, you have vertical spacing control as miniscule as half the thickness of a dot! Another way of expressing the same idea is that Gemini provides a high-resolution graphics pattern of 144 dots per vertical inch. With Gemini's ultra-high horizontal resolution capability of 240 dots per inch, that makes for a powerful graphics tool, no matter how you look at it.

In Figure 6-5, you'll see 1/144th-inch resolution, as well as the appearance of other "leadings" (pronounced "leddings"). Leading or line spacing is the amount of space set between lines of type.

Figure 6-5 presents sample leadings of 1/144th inch to approximately 1/5th inch. If you want to run the sample program that produces Figure 6-5, it is presented in BASIC. If you want to run this program, the following paragraphs discuss the "what's and "why's."

## A Sample Program that Demonstrates Vertical Spacing

The program presented in Figure 6-5b is short but takes awhile to print. In the program, the first line prints a heading, a built-in title for your graphics. Line 20 starts a loop from 1 to 30: This loop controls the vertical spacing.

Line 30 may look strange, but it reveals an important part of the limitations of BASIC. Although the language is easy to use, it has a few quirks. In this case, a CHR$ code is intercepted by BASIC and used as a BASIC control code that is then sent to the printer.

CHR$(13) is a code for a carriage return. Microsoft BASIC can't resist inserting a line feed, CHR$(10), behind every CHR$(13) that it handles *en route* to the printer. The computer language doesn't know that this particular CHR$(10) is the final character in the ESC "3" *n* command that tells Gemini to change the "leading" (line feed) to 13/144ths inch. There are ways to get around the problems (using the POKE statement, for example). For this program, we will bypass the problem and go to line 110 when the loop counter (I) equals 13.

Line 40 reveals that the variable M is used to control the number of times line 90 is executed. When I is small, M is large. When I is greater than 12, M is fixed to 2. Line 50 sets the vertical line spacing (you may recall that ESC = CHR$(27), "3" = CHR$(51)). So, in this program, line 50 sets the vertical spacing to 1/144 inches.

Line 60 starts a loop that uses the value of M to count 25 times when I equals 1, and counts 3 times when I is greater than 12. Line 70 prints a message telling us the value of the line spacing on the next-to-last line of the M loop. Line 80 spaces over the printing done by line 90 to column 30 by using the BASIC TAB statement.

Line 90 is where the action is. This line is similar to line 140 in the first reference program and prints 40 characters; its ASCII code is 241. This character is a six-dot straight line that is part of the block graphics set.

Line 100 and 110 end the J loops, and line 120 resets the printer and ends the program.

## Discussion of the Vertical Spacing Program

Let's start by looking at how close the dots are vertically in Figure 6-5. Remember that the normal characters are seven dots high, which is 14/144ths (7/72nds) of an inch. Therefore, if you wanted to print text, say in all uppercase letters, with each line resting atop the next, you'd set the vertical spacing at 14/144th of an inch.

When we use block graphics (where each block is 6 dot-spaced high), we set the vertical spacing to 6/72 (or 12/144) inches.

The maximum value of *n* allowed for the vertical spacing control code is 127. The result is that almost one inch (127/144ths) for the ESC "3" *n* code, and roughly 1-3/4 inches (127/72nds) for ESC "A" *n*.

In the sample program, we stopped I at 30, so the printout would fit on a standard 66-line sheet of sprocket-fed paper. If you wish to, you can replace the loop limit on line 20 with 127, and run the program again. You might also try replacing the CHR$(51) with CHR$(65) on line 50 and run the program using the *n*/72-inch vertical line spacing mode.

## High-Resolution Vertical Spacing Using the ESC 3 *n* Control Code

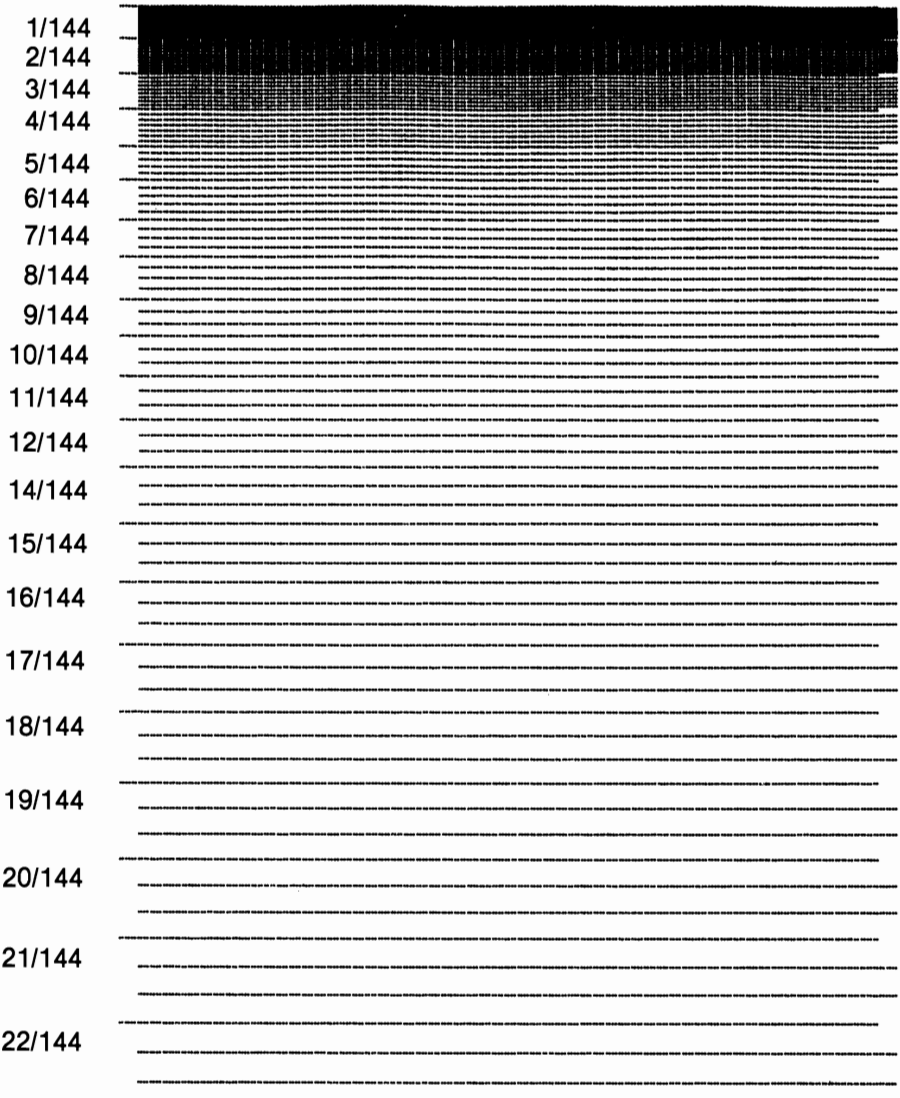



*Figure 6-5a. This printout (actual size) shows the vertical spacing that you can control using the ESC 3 n code. This printout was created using the BASIC program shown at the bottom of Figure 6-5b (a continuation of this figure).*

Inch Spacing

23/144

24/144

25/144

26/144

27/144

28/144

```
10 LPRINT TAB(30);"High Resolution Vertical Spacing "
15 LPRINT TAB(30);"Using the ESC 3 n Control Code "
20 FOR I = 1 TO 30
30 IF I = 13 THEN GOTO 110
40 M = INT ( 144 / ( 6 * I ) ) : IF M < 2 THEN M = 2
50 LPRINT CHR$(27); CHR$(51);CHR$(I);
60 FOR J = 0 TO M
70 IF J = M-1 THEN LPRINT I ; "/144 INCH SPACING";
80 LPRINT TAB(30);
90 FOR N = 1 TO 40: LPRINT CHR$(241); : NEXT N: LPRINT
100 NEXT J
110 NEXT I
120 LPRINT CHR$(27);"@": END
```

*Figure 6-5b. This continuation of Figure 6-5a depicts a more "open" spacing, that you might use for printing drafts of documents that need room for letting the reviewer manually insert changes. This printout was produced by the BASIC program listed above, written for the IBM-PC. Consult the appendices to learn how to adapt this program to other computers.*

## INTRODUCTION TO DOT GRAPHICS

When we enter into the bit-graphics mode, we have complete control over the firing of each individual pin in the print head. The graphics mode opens up the full potential of the Gemini printer by allowing us to create any image we can imagine. When printing dot graphics, Gemini can produce an ultra-high-resolution image (at 240 x 144) of over thirty thousand dots per square inch!

There are 80 characters, for example, in an 8-inch-wide line of text set in 10 CPI. For the same line width, you can print 136 condensed-width (17 CPI). As a point of comparison, you could fire up to 480 dots per line (that's 60 dots per linear inch) in the "low" resolution graphics mode. In the ultra "high" resolution graphics mode, you could print graphic images with a precision of 1920 dots per line (four times the number of "firings" per line-inch as in "low" resolution). "Dot's" a lot of dots.

Fortunately, we don't have to fill all the columns on every line, any more than a painter must apply all available paints to a single canvas.

After telling the printer to enter the graphics mode, we then define how many columns of dots we want to print in a line. We can send Gemini a string of instructions that tell which dots we want to fire in each column. A line-feed command can be used to conclude the graphics command sequence. We can repeat the graphic-mode commands, if that's the goal, to repeat the graphic mode commands over and over or to fill an entire page with dot image graphics.

### Graphic Control Codes

The low-resolution graphics mode is entered by sending Gemini a four-character control code:

ESC "K" *n*1*n*2

where,

ESC is the character represented by ASCII code CHR$(27)

"K" is the character for ASCII code CHR$(75), which signals Gemini to put low resolution into effect.

***Note:*** If we used "L" or CHR$(76) instead of "K," we would be asking Gemini for high-resolution graphics. If we used "y" or CHR$(121), we would be asking for high-resolution graphics with double speed, and if we used "z" or CHR$(122), we would be asking for ultra-high-resolution graphics at normal speed.

*n*1 and *n*2 represent the set of ASCII codes that define how many columns of graphics to print on a line. If you want to print "x" number of columns of graphics on a line, then:

*n*1 is the remainder of x/256, and
*n*2 is the quotient of x/256

***Note:*** For those of you who are computer-wise and curious:

*n*1 = x MODULO 256, and *n*2 = INTEGER of x/256

What does all this mean to the rest of us? The following table might help.

| **If the number of columns,x, ranges from:** | **then, *n*1 is** | **then *n*2 is** |
|---|---|---|
| 1 to 255 | x | 0 |
| 256 to 511 | x — 256 | 1 |
| 512 to 767 | x — 512 | 2 |
| 768 to 1023 | x — 768 | 3 |
| 1024 to 1279 | x — 1024 | 4 |
| 1280 to 1535 | x — 1280 | 5 |
| 1536 to 1791 | x — 1536 | 6 |
| 1792 to 1920 | x — 1792 | 7 |

***Remember:*** *Gemini-10X can only print up to 480 columns in low-resolution mode, 960 columns in high-resolution mode, and 1920 columns in ultra-high-resolution mode. The Gemini-15X can print 816 columns in low resolution, 1632 in high-resolution, and 3264 in ultra-high-resolution.*

Gemini requires that precisely the correct number of characters you specify with *n*1 and *n*2 will follow immediately after the *n*2 control code. If there are too many (or too few) characters, unexpected and crazy things will happen.

The high-resolution graphics mode is entered by sending Gemini the control code:

ESC "L" *n*1*n*2, which selects the 120 dots-per-inch column scan bit image graphics mode.

But Gemini's graphics capabilities don't stop there. The Gemini-10X/15X, with control codes ESC "y" *n*1*n*2 and ESC "z" *n*1*n*2, will take you two steps beyond high-resolution graphics, and allow you to print faster and with even greater density.

ESC "y" *n*1*n*2 commands the printer to print dual-density graphics with double-speed. In other words, your result will be high-resolution graphics (as if you had used ESC "L") but they will be printed twice as fast!

ESC "z" *n*1*n*2 puts Gemini in the quadruple-density graphics mode (even more dots per inch). The result of this ultra-high-resolution mode is the capacity to print 1,920 dots in one line on the Gemini-10X, and 3264 dots in one line on the Gemini-15X. (See the explanation of *n*1 and *n*2 for ESC "K" to define how many columns of dots you want to print in a line.)

## Firing the Pins

There are nine pins in the Gemini print head. In the graphics mode, you can control the upper eight. For those of you with computers that can only send seven bits (Apple II and some others), you control the top seven pins. Let's look at the print head again and label the pins as follows:

| Print Head Pin Number | Print Head (top) | 8-Bit Interface | 7-Bit Interface |
|---|---|---|---|
| 1 | | 128 | 64 |
| 2 | | 64 | 32 |
| 3 | | 32 | 16 |
| 4 | | 16 | 8 |
| 5 | | 8 | 4 |
| 6 | | 4 | 2 |
| 7 | | 2 | 1 |
| 8 | | 1 | not used |
| 9 | | not used | not used |

Based on the chart presented above, if you had a 7-bit computer and wanted to fire pins number 1 and 4 simultaneously, you would send CHR$(64+8). In other words, you'd send CHR$(72): Therein is the pattern. Thus, if you wanted to fire all the pins, using an 8-bit computer, you'd send:

CHR$(128 + 64 + 32 + 16 + 8 + 4 + 2 + 1) = CHR$(255)

| Control Code | Binary Representation | Pin Fired |
|---|---|---|
| CHR$(128) | 1 0 0 0 0 0 0 0 | #1 |
| CHR$( 64) | 0 1 0 0 0 0 0 0 | #2 |
| CHR$( 32) | 0 0 1 0 0 0 0 0 | #3 |
| CHR$( 16) | 0 0 0 1 0 0 0 0 | #4 |
| CHR$( 8) | 0 0 0 0 1 0 0 0 | #5 |
| CHR$( 4) | 0 0 0 0 0 1 0 0 | #6 |
| CHR$( 2) | 0 0 0 0 0 0 1 0 | #7 |
| CHR$( 1) | 0 0 0 0 0 0 0 1 | #8 |
| CHR$( 0) | 0 0 0 0 0 0 0 0 | none |
| CHR$(255) | 1 1 1 1 1 1 1 1 | all 8 |
| CHR$( 72) | 0 1 0 0 1 0 0 0 | #2 & #5 |

You computer aficionados may have already figured out that the high-order bit of the character controls the top pin, the next bit controls the number-two pin, etc.

Okay, now that you know how to fire the pins, what code do you think will fire pins no. 5, 6, and 8 with an 8-bit interface (per the above chart)? How about CHR$ (8+4+1), or CHR$(13)? That may seem like a good answer, but (in the words of Hamlet) "ay, there's the rub."

CHR$(13) causes BASIC to automatically insert a line feed, CHR$(10), behind it. These BASIC programs were written several years ago, before Gemini came along. Gemini is smart enough to use CHR$(13) for a purpose other than initiating a carriage return; if only BASIC were as smart.

One solution to the above problem is to design your graphics programs so that they only use pins numbers 1 through 7 for 8-bit computers, and numbers 1 through 6 for 7-bit. Because the bottom-most pin that normally can be used is fired by CHR$(1), all commands for the other usable pins will be even-numbered (whatever their combination of numbers to fire whatever combination of pins). With all the codes even-numbered, there's no need to concern ourselves about unlucky old CHR$(13). You might want to think about it for a while. There are other ways around the problem, but the even-number approach works (even if there seems to be something odd about it).

## Example Graphics Program

Now that we've learned how to send the graphics controls to Gemini, let's construct a simple character to test low-resolution printing. This program will show you one approach to creating graphic images. In the next chapter, we'll consider more advanced techniques.

The first step in generating a sample program is to decide where to position the dots. One way is to start with a pencil and a piece of paper, especially paper with square grids printed on it. We will imagine that one dot will cover one square.

Next, you should construct the character you wish to "draw" and label the rows and columns. Figure 6-6 shows one such process, using two lines of seven rows each.

ROUGH SKETCH

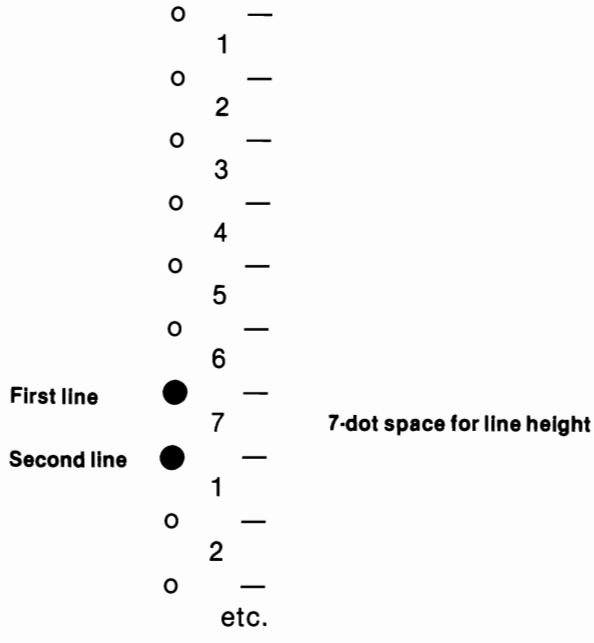
| | Firing Codes | Pin No. | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | 128 | 1 | | | | | ● | ● | ● | ● | ● | ● | | | | |
| | 64 | 2 | | | | ● | | | | | | | ● | | | |
| | 32 | 3 | | | ● | | | | | | | | | ● | | |
| Line 1 | 16 | 4 | | ● | | | | | | | | | | | ● | |
| | 8 | 5 | ● | | | | ● | ● | | | ● | ● | | | | ● |
| | 4 | 6 | ● | | | | ● | ● | | | ● | ● | | | | ● |
| | 2 | 7 | ● | | ● | | | | | | | | | ● | | ● |
| | 128 | 1 | ● | | ● | | | | | | | | | ● | | ● |
| | 64 | 2 | ● | | ● | ● | | | | | | | ● | ● | | ● |
| Line 2 | 32 | 3 | ● | | ● | | ● | | | | | ● | | ● | | ● |
| | 16 | 4 | | ● | | | | ● | ● | ● | ● | | | | ● | |
| | 8 | 5 | | | ● | | | | | | | | | ● | | |
| | 4 | 6 | | | | ● | | | | | | | ● | | | |
| | 2 | 7 | | | | | ● | ● | ● | ● | ● | ● | | | | |
| Line 1 firing code: | | | 14 | 16 | 34 | 64 | 140 | 140 | 128 | 128 | 140 | 140 | 64 | 34 | 16 | 14 |
| Line 2 firing code: | | | 224 | 16 | 232 | 68 | 34 | 18 | 18 | 18 | 18 | 34 | 68 | 232 | 16 | 224 |

*Figure 6-6. One approach to creating your own graphics is first to sketch the figure on graph paper, then code the dots for programming.*

In our example, we want to start printing in dot column 100, and print to column 114. Therefore, the total number of characters we will send for each of the two rows is 114. Using our handy table, we know that *n*1 = 114 and *n*2 = 0. Therefore, the control code we send to Gemini to put the printer in the low-resolution mode, ready to print our 14 columns, is as follows:

```
LPRINT CHR$(27); "K"; CHR$(114); CHR$(0);
```

Before you start firing the pins, be sure that the top of the second row of dots is one-dot's space below the bottom of the first row. (It's like having one picture in two frames and wanting the frames lined up close together.) You are using seven printing dots, and each dot is 1/74 of an inch below the one above it. Perhaps this illustration will make it clearer.



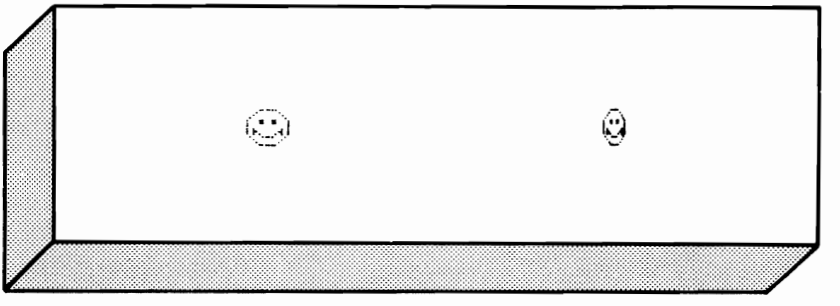

Our line feed spacing should be 7/72 inch. We have a special line-feed command for precisely that spacing:

ESC "1" or, CHR$(27) CHR$(49)

Now, it's just a matter of entering the program.

```
10 RESTORE
15 WIDTH "LPT1:",255
20 LPRINT CHR$(27);"@";CHR$(27);"1"
30 FOR L = 1 TO 2
40 LPRINT CHR$(27);"K";CHR$(114);CHR$(0);
50 FOR SPACE = 1 TO 100: LPRINT CHR$(0);: NEXT SPACE
60 FOR COLM = 1 TO 14 : READ C : LPRINT CHR$(C);: NEXT COLM
70 LPRINT : NEXT L
80 DATA 14,16,34,64,140,140,128,128,140,140,64,34,16,14
90 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
100 LPRINT CHR$(27);"@";LPRINT:LPRINT:LPRINT
```

Let's run the program and see if we can get a smile both on the paper and on the face of the Gemini operator who's creating dot matrix graphics.

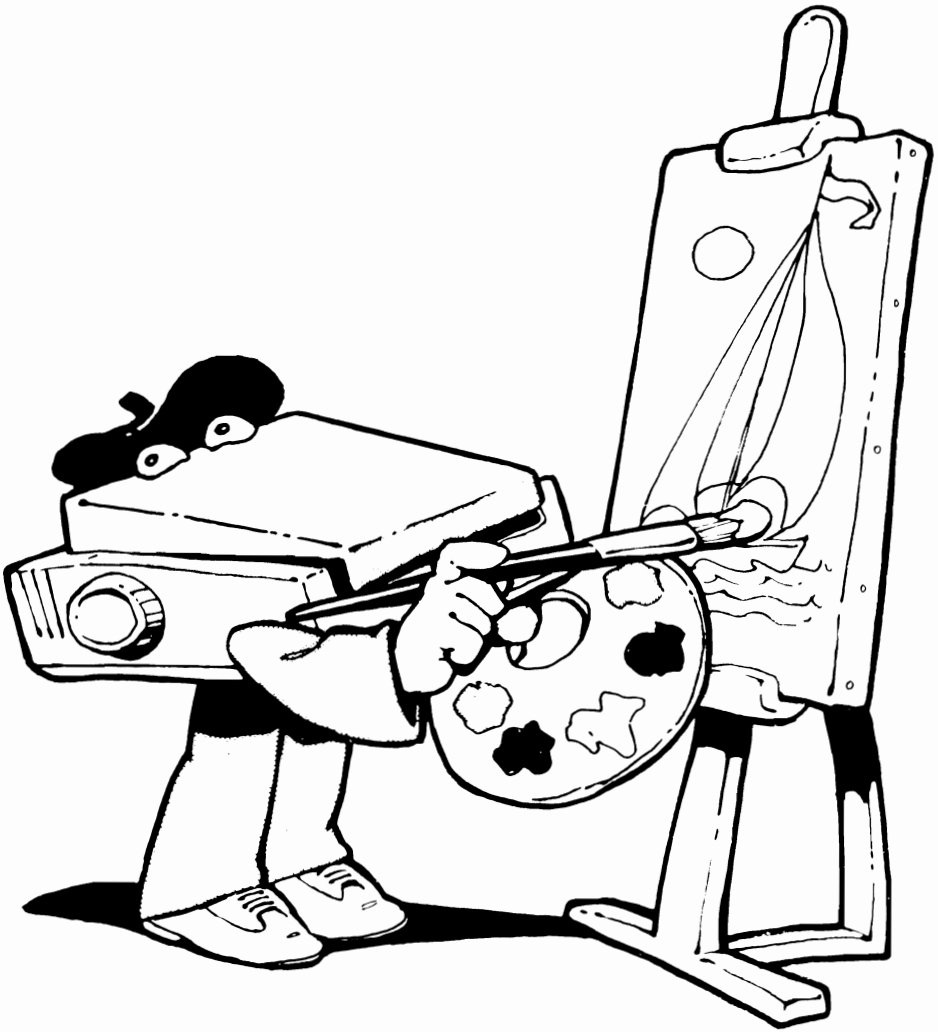


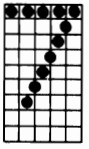
Now, if you want to try a little something different, change line 40 to:

40 LPRINT CHR$(27); "L"; CHR$(114); CHR$(0);

and run the program again. High-resolution packs the dots twice as close together, horizontally. For the same size drawing, you would need twice as many dots. Therefore, you'd have to send twice as many print head control characters to Gemini.

By this time, you probably have a good idea of how Gemini works and how to make use of its extensive features. You might want to take some time now and explore your own ideas, based on the information presented thus far. In the next chapter, we'll consider the use of special symbol characters, block graphics, and some advanced graphic techniques.





# GEMINI'S FULL RANGE OF PRINTING CAPABILITIES

This chapter builds on the foundation of Chapters 5 and 6 and demonstrates how you can put Gemini to work for you.

As with the other chapters, this one presents programs written in Microsoft BASIC for the IBM-PC. If you have an Apple, Atari, Commodore, Osborne, or TRS-80, you should consult the Appendices for those computers at the back of this manual. If you have a computer other than one of those just mentioned, look in the appendix of the computer most like the one you have. To learn more about running programs in BASIC, consult the user's manual for your computer.

This chapter begins with a discussion of using Gemini's expanded character set. The middle of this chapter considers low-resolution graphics, with sample programs. Chapter 7 concludes with some tips on producing high-resolution graphics.

## FOREIGN LANGUAGE CHARACTERS

Gemini can converse in the English language, as well as in most of the Romantic languages (French, Spanish, Italian, German, etc.). By the way, the "Romantic" languages are so-named because of the Latin influence of the Romans, not because vacationing Americans find the European locals amorously irresistable. Swedish and Danish are also available.

### Some Characters for French

Consider, the following characters and their ASCII code equivalents:

| à | é | è | ú | ç |
|---|---|---|---|---|
| CHR$(193) | CHR$(219) | CHR$(221) | CHR$(220) | CHR$(194) |

To incorporate these characters, you can follow the same patterns you've already learned for using BASIC.

For example,

```
LPRINT "Plus"; CHR$(194);
  "a change plus c'est la me";
LPRINT CHR$(8); CHR$(94); "me chose."
```

will print:
Plus ça change plus c'est la même chose.
("The more things change, the more they remain the same.")
We used the CHR$(194) character to print the cedilla ("ç"). For the "ê," type the letter "e" and then backspace — CHR$(8) — followed by CHR$(94) for the circumflex. C'est très facile, n'est-ce pas?

## Spanish

If you wanted to print: Se habla español
("Spanish spoken"),
you could input the following controls:

```
LPRINT "Se habla espan"; CHR$(8) CHR$(126); "ol"
```

## German

German text, when set in modern type, requires only the umlauted ( ¨ ) vowels.
For example, to print:
für den eigen en Bedarf
("for personal consumption")
You would enter:

```
LPRINT "f"; CHR$(216); "r den eigen en Bedarf"
```

## Foreign Language Characters: Summary

Gemini now provides you with even more non-English special characters to allow you to communicate in most European languages.

There are 2 ways in which to access this International character set:

1. By using the ESC 7*n* code where:

| | | | |
|---|---|---|---|
| USA: | *n* = 0 | England: | *n* = 1 |
| Germany: | *n* = 2 | Denmark: | *n* = 3 |
| France: | *n* = 4 | Sweden: | *n* = 5 |
| Italy: | *n* = 6 | Spain: | *n* = 7 |

2. By using DIP switch 1 (refer to Figure 4-6)

| DIP Switch Pin | | | Country |
|---|---|---|---|
| **1-6** | **1-7** | **1-8** | |
| on | on | on | USA |
| off | on | on | England |
| on | off | on | Germany |
| off | off | on | Denmark |
| on | on | off | France |
| off | on | off | Sweden |
| on | off | off | Italy |
| off | off | off | Spain |

## MONETARY COMMUNICATION

You've probably heard the expression: "Money talks." Your Gemini can speak the language of business, as well:

| Dollars | Cents | Pounds | Francs | Yen |
|---|---|---|---|---|
| $ | ¢ | £ | *f* | ¥ |
| CHR$(36) | CHR$(212) | CHR$(195) | CHR$(223) | CHR$(208) |

## SCIENTIFIC AND MATHEMATIC NOTATION

Science, engineering, and mathematics have their own "languages" to communicate technical ideas. The versatile Gemini can meet the challenge of modern science by printing special characters as required.

### Mathematical Operations

Beyond superscripting (initiated by ESC "S" 0) and subscripting (ESC "S" 1), the following are Gemini's symbols for mathematical operations:

| plus | minus | greater than | less than | plus or minus |
|---|---|---|---|---|
| + | – | > | < | ± |
| CHR$(43) | CHR$(45) | CHR$(62) | CHR$(60) | CHR$(188) |

| divided by | | equals | times | tilde |
|---|---|---|---|---|
| ÷ | / | = | × | ~ |
| CHR$(191) | CHR$(47) | CHR$(61) | CHR$(190) | CHR$(126) |

You can also combine characters, using the backspace command:

LPRINT CHR$(126); CHR$(8); CHR$(95)

produces: ≃, the symbol for "approximately equal to."

## Printing Equations

Gemini provides three different types of "grouping" symbols for use in mathematical expressions:

| ( | ) | [ | ] |
|---|---|---|---|
| CHR$(40) | CHR$(41) | CHR$(91) | CHR$(93) |

| { | } |
|---|---|
| CHR$(123) | CHR$(125) |

The character set of Greek symbols, often used in scientific and mathematical equations, include:

| sigma | | pi | omega | |
|---|---|---|---|---|
| Σ | σ | π | Ω | ʊ |
| CHR$(184) | CHR$(185) | CHR$(187) | CHR$(182) | CHR$(183) |

| phi | theta | mu | beta |
|---|---|---|---|
| ∅ | θ | μ | β |
| CHR$(178) | CHR$(179) | CHR$(197) | CHR$(217) |

Appendix H, at the back of this manual, lists all special characters. If you take a moment and look over the character set, you'll find the symbols for the parallel lines, functions, angstroms, degrees (in C and F), and other characters that serve special purposes in scientific notation.

## SPECIAL SYMBOLS FOR PLOTTING DATA

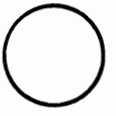
For those of you who would like to use Gemini to plot data and graphs, a nice set of plotting symbols are provided:

| ○ | △ | ▽ | ▷ | ◁ | ◇ | ◆ | □ |
|---|---|---|---|---|---|---|---|
| CHR$(168) | (169) | (170) | (171) | (172) | (173) | (174) | (175) |

ASCII code 160, 161, 162, and 163 provide the four segments of a "two-line" circle:

```
LPRINT CHR$(163); CHR$(161); CHR$(27); "A"
    CHR$(6)
LPRINT CHR$(162); CHR$(160); CHR$(27); "2"
```

This little program will produce:



You can use two arcs from the circle to produce a different symbol:

```
LPRINT TAB (2);CHR$(163);CHR$(27);"A";CHR$(6)
LPRINT CHR$(160)
```

The result is a two-line symbol that serves as the scientific notation for the integration operation:

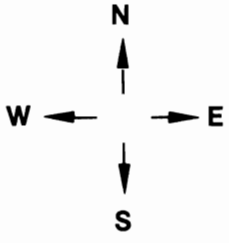
The four directional arrows:

CHR$(164) CHR$(165) CHR$(166) CHR$(167)

can be utilized to produce a cartographic (map-making) symbol.

```
LPRINT TAB (20); "N"; CHR$(13);
      TAB (20); CHR$(164); CHR$(13);
      TAB (16); "W"; CHR$(166); CHR$(32);
      TAB (22); CHR$(167); "E"; CHR$(13);
      TAB (20); CHR$(165); CHR$(13);
      TAB (20); "S"
```

The result is:




## BLOCK GRAPHICS

The block graphics, ASCII codes 225 to 255, are a series of squares, triangles, and intersecting lines. When they are put together, they form simple graphic designs and decorative embellishments.

The 6 x 6 matrices are designed to print continuously (meaning "next to each other"), so Gemini does not add any extra spacing between these characters, as it does for the alphanumerics.

The Gemini can now print in a bidirectional mode (ESC U O), which cancels unidirectional printing and returns to the faster (standard) bidirectional printing mode or it can print in unidirectional mode (ESC U 1). In this mode all subsequent lines are printed in unidirectional mode. This mode is invoked by the super/subscript mode and by the graphic mode.

### The Building Blocks

In the last chapter, we used block graphics to create oversized letters in Figure 6-4. Let's consider that example again, this time with discussion of the program listing.

```
10 LPRINT CHR$(27);CHR$(65);CHR$(6)
15 LPRINT CHR$(27);CHR$(85);CHR$(1);
20 RESTORE
30 FOR J = 1 TO 9
40 FOR I = 1 TO 12
50 READ A
60 LPRINT CHR$(A);
70 NEXT I
80 LPRINT
90 NEXT J
100 DATA 239,224,224,224,239,224,  224,224,224,224,224,224
110 DATA 239,224,224,224,239,224,  224,224,224,224,224,224
120 DATA 239,224,224,224,239,224,  239,224,224,224,239,224
130 DATA 239,224,239,224,239,224,  234,233,224,234,233,224
140 DATA 239,234,239,233,239,224,  224,239,224,239,224,224
150 DATA 239,239,224,239,239,224,  224,234,239,233,224,224
160 DATA 234,233,224,234,233,224,  224,224,239,224,224,224
170 DATA 224,224,224,224,224,224,  224,234,233,224,224,224
180 DATA 224,224,224,224,224,224,  224,239,224,224,224,224
190 LPRINT CHR$(27);CHR$(64);END
```

*Figure 7-1. Block graphics can produce simple output, like this, and can "draw pictures" as well.*

Line 10 of the sample program in Figure 7-1 sets consecutive line feeds to 6/72 = 1/12 inch. This ensures that the 6 x 6 block-graphic characters will stack on top of each other without unwanted vertical space between them.

Line 15 sets your printer to unidirectional print mode (as in Figure 7-1) ensuring vertical alignment.

Line 30 is a loop-over. In this example, the "W" and "y" are nine lines high.

Line 40 is loop-over for spaces. Each character is six spaces ("blocks") wide. For our example, we've created "oversized" letters that have the same height-to-width proportions as the standard character set. Because there are two letter-characters being printed they represent 12 spaces total width, with an extra column inserted as a "spacer" between them for legibility.

Thus, you can see that in creating this big "Wy," we've followed the same configuration as standard matrix characters, but each small "dot" on the 9 x 9 has been replaced with a "solid block" of 6 x 6's.

Line 50 reads the ASCII character codes stored in the data statement. Line 60 prints the data.

Line 80 inserts a line feed after each line. The data statements on lines 100 to 180 are arranged so that the ASCII code for each 6 x 6 symbol used to create "Wy" are in the correct location.

CHR$(239) is a solid block of 6 x 6 graphics. CHR$(224) is a blank. CHR$(234) and CHR$(233) are 3-by-6 characters which are used to represent "big dots" that we placed in positions half-way between the "normal big-dot columns."

*Why?* you might ask. Ours is not to wonder *why*, but to explain *how*. A few minutes of reviewing Appendix K and some sketches on "quad" paper should convince you that studying Appendix K and making some preliminary sketches on graph paper are probably the best first steps you can take in designing your own block graphic projects.

This sample program demonstrates some of what block graphics can accomplish, but why limit ourselves to the ASCII code numbers 224 to 239? In the next program, we'll set sail for more challenging applications.

## Creating Graphic Drawings

Figure 7-2 presents a sailboat, "drawn" with 6 x 6 block graphics. Also included in Figure 7-2 is the program listing. If you want to take a few minutes to enter and run the program, go ahead. On the other hand, you could learn some points about block graphics if you just read over the listing and follow the discussion.

As in the program in Figure 7-1, line 10 in this program sends a message to set the line feed at 6/72 inches.

Line 15 is used to set the printer to unidirectional print mode. Line 20 is used to loop over the section of code that prints the sailboat. Line 30 is used to control the font pitch (the spacing between dots). The first time, K = 1, so line 30 tells Gemini to set 10 CPI spacing. The second time the boat is printed, K = 2, and line 30 tells the printer to use 12 CPI spacing. The third time, K = 3, and line 30 calls for 17 CPI.

```
10 LPRINT CHR$(27);CHR$(65);CHR$(6);CHR$(15)
15 LPRINT CHR$(27);CHR$(85);CHR$(1);
20 FOR K = 1 TO 3
30 LPRINT CHR$(27);CHR$(66);CHR$(K)
40 RESTORE
50 FOR J = 1 TO 9
60 FOR I = 1 TO 12
70 READ A
80 LPRINT CHR$(A);
90 NEXT I
100 LPRINT
110 NEXT J
120 DATA 224,224,224,224,224,224,  224,224,224,245,224,224
130 DATA 224,224,224,224,224,224,  224,224,252,245,224,224
140 DATA 224,224,224,245,224,224,  224,252,239,245,224,224
150 DATA 224,224,252,245,224,224,  252,239,239,245,224,224
160 DATA 224,252,239,245,224,252,  239,239,239,245,224,224
170 DATA 252,239,239,245,252,239,  239,239,239,245,224,224
180 DATA 241,241,241,249,241,241,  241,241,241,249,224,224
190 DATA 239,239,239,239,239,239,  239,239,239,239,239,251
200 DATA 253,239,239,239,239,239,  239,239,239,239,251,224
210 LPRINT : NEXT K
220 LPRINT CHR$(27);CHR$(64);:END
```

*Figure 7-2. Block graphics created this fleet of sailboats, using the simple program shown.*

Line 40 is a BASIC command that tells the next READ statement to start reading at the beginning of the data.

The data statements have been changed in Figure 7-2 from what they were in Figure 7-1, or else we would have produced another "Wy" instead of the sailboat. You can examine the data to see what characters are used, and with what results. You might want to take a sheet of quad paper and mark off 12 squares by 9 squares; then write the symbol for each character in the corresponding position.

Note that we can control the aspect ratio (height to width) of the block graphic characters in the same manner we control the standard characters.

## Upgrading Our Design

Before our ship sails, let's add some portholes. Replace line 190 with:

190 DATA 239,175,239,175,239,175,239,175,239,175,239,251

When you run the program again, you'll produce the sailboats shown in Figure 7-3. The lesson learned is that you can mix the 6 x 6 graphics with standard 9 x 9 characters.

These special characters in the block-graphics set can be very useful in creating bar graphs, your organization's logo, even detailed drawings. You could spend months experimenting with the block-graphics set and still not exhaust the possibilities. But before you embark on such an adventure, consider dot graphics, which happens to be the next area of discussion.

## GEMINI'S DOT GRAPHICS

As we learned in Chapter 6, dot graphics commands allow you to control the firing of individual pins in the print-head — in short, these commands give you total control over your printer. That's the good news. The rest of the news is that you have to tell Gemini where to place, exactly, every single dot. And there can be a lot of dots in a dot graphics program. That's like being given ocean-front property, if you put every grain of sand, one by one, on the beach. But wait, there's more good news.

The computer you have interfaced with Gemini is a powerful tool when it comes to moving large amounts of data. All you need to do is tell your computer where you want dots placed, and it will do most of the rest. It's like getting the offer on the beach house and having access to a dumptruck to help install the mountains of sand.

In the conclusion to Chapter 7, we'll develop a program that will allow you to "draw" on an 8-1/2-by-11-inch sheet of paper. With the program, if you specify where you want a line to begin and end, the computer will fill-in the dots in between. It's not altogether that simple, but it's not really that hard either.

One word of warning: Memory. Bit-image graphics (constructing images from individual dots, or bits) requires generous memory capacities. In our example, we'll need at least 16K of memory. But there's even good news in that: you can cut down on the image area, as we'll explain later.

## The Concept

We are going to construct a bit-image "map" of the sheet of paper in the computer's memory. Then we're going to print that image on paper by selectively firing the dot-producing pins on the print head.

As you know, the memory in your computer functions in terms of bits, which are represented as being either "0" or "1". We'll use each bit that's part of our map to either be a dot ("1") or a blank space ("0"). Our picture will be 360 dots wide (at 60 dots per linear inch) and 270 dots in height.

We are going to hold 6 bits in each character to avoid problems that you might have with the carriage-return command and to let people with either 7-bit and 8-bit computers participate without significant adaptation of our program. This program, like the previous ones, is written for the IBM-PC, but the appendices at the back of this manual will tell you how to adapt this program to several other computers.

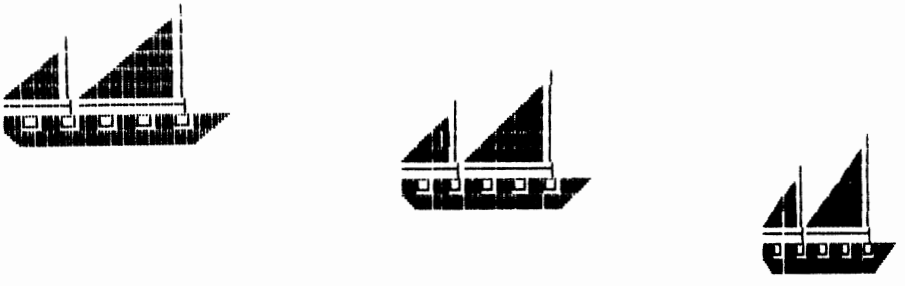


*Figure 7-3. In this program, standard characters (the "o's" in portholes) are mixed with 6 x 6 block graphics on the same line.*

## The Program

The program listed in Figure 7-4 is made up of subroutines. A sample output is provided (Figure 7-5) to illustrate the program's capabilities.

So, before exercising our brains with explanations, let's type in the program as written, and RUN it.

You may feel like it takes a long time to run the program. The

extra wait is part of the price for BASIC's simplicity. In addition, detailed graphics like this involves massive data manipulation.

If for any reason you didn't get an output that looks like it was supposed to look (compare with Figure 7-5), then turn Gemini "off," reset it by turning it back "on," enter LLIST, and compare the resultant printout to see where you went wrong. After you fix the problem, RUN the program.

## Analyzing the Program

This program will take about 3 minutes to run. It generates 900 bytes of data that will be printed using the ESC "K" function. The first part of the program will initialize several constants as well as an array to store our data. The array MASK%() contains the values for each dot. Notice that we are using only six dots in this program. This is to make sure that this program will work with all computers, even those with 7-bit interfaces. LX and LY provide a scale for our image and XFAC and YFAC relate the scale to the overall size of the image. RAD is the radius of the shape; X1 and Y1 are the starting points for the curve.

The subroutine that starts at line 1000 will plot the points around the curve and will call the subroutine starting at line 3000 to actually plot the line from $X_1$ $Y_1$ to $X_2$ $Y_2$. This subroutine will call yet another routine starting at line 4000; this routine actually "turns on" the dots. It stores the element in BIT%() and then uses the values in MASK%() to turn on the proper dots by use of an OR function.

Note: Some of the computers used in this book do not have an OR function; therefore, some of the programs use a machine language subroutine that will simulate our OR function. This routine is created by "POKE-ing" data into memory.

The last subroutine starts at line 2000. This subroutine will get the data and send it to your Gemini printer. The routine starting at line 2020 through 2110 uses a loop to go through the data. It will print 12 lines and each line contains 375 dots. Data will be sent to the printer one line at a time. The shape we plot will illustrate our ability to plot mathematical curves. Since we need to generate high quantities of data through our equations, please note that it will take a few minutes to use this program.

```
10 CLS:PRINT "":PRINT "":PRINT "":PRINT ""
20 PRINT "THIS PROGRAM TAKES ABOUT 3 MINUTES TO RUN"
30 PRINT "PLEASE TURN ON YOUR PRINTER AND STAND  BY"
100 DIM BIT%(76,12)
110 MASK%(1)=128:MASK%(4)=16
120 MASK%(2)=64 :MASK%(5)=8
130 MASK%(3)=32 :MASK%(6)=4
140 LX=20:LY=20
150 XFAC=72/LX:YFAC=75/LY
1000 REM PLOT CURVE
1010 RAD=9
1020 X1=19:Y1=10
1030 FOR ANG=0 TO 360 STEP 10
1040 R1=ANG*6.28/360
1050 X2=RAD*COS(R1)+10:Y2=RAD*SIN(R1)+10
1060 GOSUB 3000
1070 NEXT ANG
1080 FOR ANG=0 TO 360 STEP 15
1090 R1=ANG*6.28/360
1100 R2=(ANG+150)*6.28/360
1110 X1=RAD*COS(R1)+10:Y1=RAD*SIN(R1)+10
1120 X2=RAD*COS(R2)+10:Y2=RAD*SIN(R2)+10
1130 GOSUB 3000
1140 NEXT ANG
2000 WIDTH "LPT1:",255
2010 LPRINT CHR$(27);"A";CHR$(6)
2020 FOR ROW=0 TO 11
2030 A$=""
2040 LPRINT CHR$(27);"K";CHR$(119);CHR$(1);
2050 FOR COL=1 TO 75
2060 FOR LL=1 TO 5
2070 LPRINT CHR$(BIT%(COL,ROW));
2080 NEXT LL
2090 NEXT COL
2100 LPRINT
2110 NEXT ROW
2120 LPRINT
2130 END
3000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
```

*Figure 7-4a. This BASIC program will allow you to create your own low- and high-resolution bit-image graphics. (continued)*

```
3010 XL=X2-X1:YL=Y2-Y1
3020 NX=ABS(XL*XFAC):NY=ABS(YL*YFAC)
3030 IF NX<NY THEN NX=NY
3040 NS%=INT(NX+1)
3050 DX=XL/NS%:DY=YL/NS%
3060 FOR I=1 TO NS%
3070 X1=X1+DX:Y1=Y1+DY
3080 GOSUB 4000
3090 NEXT I
3100 RETURN
4000 REM PLOT A POINT AT X1,Y1
4010 XX=X1*XFAC:YY=Y1*YFAC
4020 COL=INT(XX)+1
4030 ROW=INT(YY/6)
4040 XIT%=INT(YY-(6*ROW))+1
4050 BIT%(COL,ROW)=BIT%(COL,ROW) OR MASK%(XIT%)
4060 RETURN
```

*Figure 7-4b. This continuation of Figure 7-4a is the rest of one approach to using bit-image graphics.*

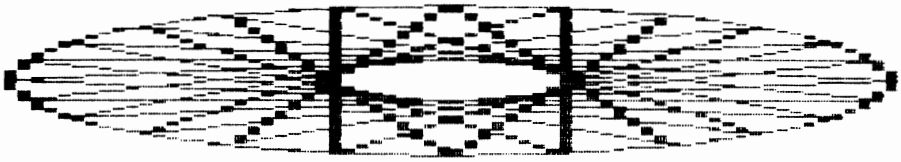


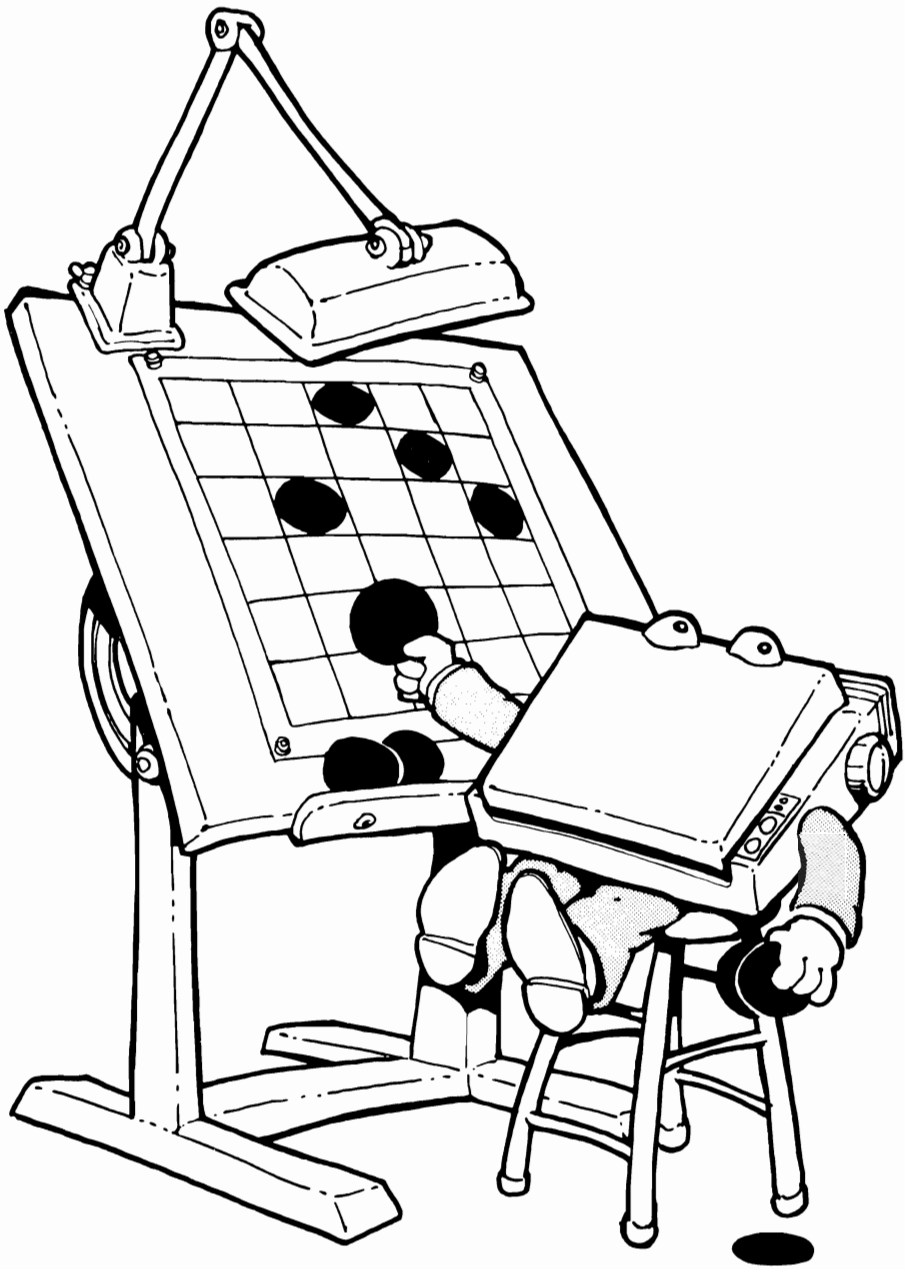
*Figure 7-5. Using the BASIC program provided in Figure 7-4, you can obtain these results.*

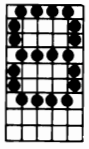
## CONCLUSION

The limits on Gemini's ability to produce readable text and graphic images are limited only by your imagination, your computer system, and your ability to properly use the functional command codes.

When it comes to "imagination" and a good computer system, you stand on your own ground. We can help you master the function codes, as demonstrated in Chapter 9.

# *Notes*





# DOWNLOADABLE CHARACTERS AND CUSTOM COMMANDS

## DOWNLOADABLE CHARACTERS AND CUSTOM COMMANDS

Beyond all the capabilities discussed so far in this manual, Gemini offers two more extremely flexible functions. If the Gemini doesn't have the characters you need already built in, you can custom design your own. (In computer jargon, this is called downloadable characters). Up to 96 downloadable characters can be memorized by the RAM of the printer. You can also use the Gemini's ability to group commands together into a single command to simplify your programming ("macro command", more jargon). This chapter explains the use of these options and gives you some practical examples from which to build.

As we learned in Chapter 3, some professions make use of special characters. For example, there are special symbols that are used by chemists, pharmacists, architects, electrical engineers, linguists, astronomers, medical researchers, and so on.

Let's start our downloadable character education by choosing a new character to create. Because the editors will be the first people to look at this text, we'll choose one that they'll feel comfortable with, the paragraph symbol, ¶. This little backward P is used in both manuscript editing and in word processing. Unfortunately, it's not one of the standard characters on most printers. Appendix K lists all of the standard characters on your printer.

To design a new character, we have to lay out which dots within a character cell (character position) should be printed. The easiest way to lay this out is to take a piece of blank paper or graph paper and mark off a block of 7 squares high by 9 squares wide, like we've done in Figure 8-1. Along the left-hand side, write the numbers 1,2,4,8,16,32, and 64 vertically. Place a dot in each square that you want a dot to appear in the character. One rule before you start placing dots, across any horizontal row, there must be a space between each dot. This isn't as bad as it seems because the actual character will be printed narrower than the graph paper shows.

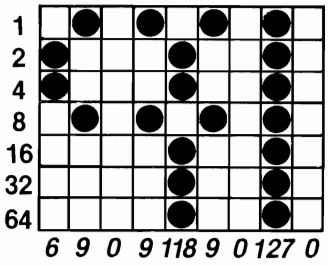



*Figure 8-1. Downloadable design for paragraph symbols with each dot given a numerical value.*

After we've placed all of the dots for our character, we need to turn them into numbers that the printer can understand. To do this, go down each column and add up the value of each dot. The numbers placed on the left-hand side are the value for each dot.

Let's take column one as an example. The first dot is in row two. Row two has a value of 2 on the left-hand side. The next dot is in row three and it has a value of 4. These are the only dots in this column, so the total value for this column is 6 (2 + 4 = 6). Write this number at the bottom of the column. After adding up the values for each column, you will have nine numbers to send to the printer. For the first example, the numbers are: 6,9,0,9,118,9,0, 127,0.

The code to define a downloadable character to the printer is:

CHR$(27) CHR$(42) CHR$(1) $c_1$ $d_1$ $t_1$ $t_2$ $t_3$ $t_4$ $t_5$ $t_6$ $t_7$ $t_8$ $t_9$

The $c_1$ is the character we wish to redefine. In the example, we'll redefine the @ symbol (CHR$(64)). When we activate downloadable characters and this character is sent to the printer, the downloaded character will be printed. The $d_1$ is the descender data. For now, we'll just leave that as zero. The $t_1$ through $t_9$ are the column totals across the bottom of the graph.

So the complete code sequence would be as follows: CHR$(27); CHR$(42); CHR$(1); CHR$(64); CHR$(0); CHR$(6); CHR$(9); CHR$(0); CHR$(9); CHR$(118); CHR$(9); CHR$(0); CHR$(127); CHR$(0).

After defining a download character, we must activate the download characters in the printer. The code to perform this trick is:

CHR$(27) CHR$(36) CHR$(1)

Let's put all this together in an example and print out a few of our new paragraph symbols.

```
10 REM  Define the Escape character
20 ESC$=CHR$(27)
30 REM  Define paragraph symbol in place of @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(6);CHR$(9);CHR$(0);
          CHR$(9);CHR$(119);CHR$(9);CHR$(0);CHR$(127);CHR$(0);
50 REM  Select downloadable characters
60 LPRINT ESC$;CHR$(36);CHR$(1);
70 REM  Print our new symbol
90 LPRINT "@@@@@@"
```

*Figure 8-2. Program listing to print:* ¶¶¶¶¶¶

Not too bad for the first try. Let's create another symbol to keep the editors happy and to demonstrate a few more things you'll need to know about downloadable characters. This time we will create the copyright symbol; a small c with a circle around it. First, we'll create the symbol on graph paper and total all the columns as we did for the paragraph symbols.

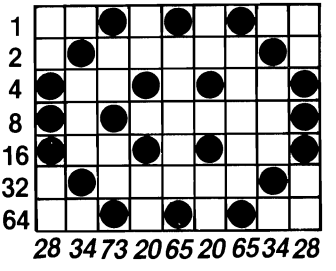



*Figure 8-3. Downloadable design for copyright symbol.*

This time we will use the downloadable feature to replace the @ symbol with the copyright symbol.

```
10 REM  Define the Escape character
20 ESC$=CHR$(27)
30 REM  Define copyright symbol in place of @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);CHR$(34);CHR$(73);
          CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR$(20);
50 REM  Select downloadable characters
60 LPRINT ESC$;CHR$(36);CHR$(1);
70 REM  Print our new symbol
80 LPRINT "Copyright @ 1983"
```

*Figure 8-4. Program listing to print:* ©

What happened? All we got was the copyright symbol and nothing else. This happened because none of the other characters in the download area (RAM) have been defined. So you need one more command when you are only replacing a few characters at a time:

CHR$(27) CHR$(42) CHR$(0)

This code copies all of the normal characters into the downloadable area (RAM) of your printer. Let's insert this into Example 2 *before* we define our copyright symbol.

```
10 REM  Define the Escape character
20 ESC$=CHR$(27)
24 REM copy characters to RAM
25 LPRINT ESC$;CHR$(42);CHR$(0);
30 REM  Define copyright symbol in place of @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);CHR$(34);CHR$(73);
          CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR$(20);
50 REM  Select downloadable characters
60 LPRINT ESC$;CHR$(36);CHR$(1);
70 REM  Print our new symbol
80 LPRINT "Copyright @ 1983"
```

*Figure 8-5. Program listing to print:* Copyright © 1983

Now things look a little more shipshape for the editors.

### Descenders

Earlier in this chapter we mentioned the descender information. When we left it as a zero you probably thought this was easy. Well, it is! We only have two choices for descender data; it's either a zero or a one. When it's a zero, nothing happens. When it is a one, all the dots that make up our character are shifted down two rows. Why do we want to perform this magic? If you look at the lowercase g in magic, you can see that the tail of the g drops down below the bottom of other characters. This is a descender. Let's take our copyright symbol example and define the @ symbol as we did before and make the "&" symbol our copyright symbol with a descender (shifted down two rows).

```
10 REM  Define the Escape character
20 ESC$=CHR$(27)
24 REM copy characters to RAM
25 LPRINT ESC$;CHR$(42);CHR$(0);
30 REM  Define copyright symbol in place of @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);CHR$(34);CHR$(73);
        CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR$(20);
44 REM  Define copyright symbol in place of &
45 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(38);CHR$(1);CHR$(28);CHR$(34);CHR$(73);
        CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR$(20);
50 REM  Select downloadable characters
60 LPRINT ESC$;CHR$(36);CHR$(1);
70 REM  Print our new symbols
90 LPRINT "@&@&@&@&@&@&@&@&"
```

*Figure 8-6. Program listing to print:* ©©©©©©©©©©©©©©©©

One more instruction you'll need before we leave downloadable characters: How to turn them off.

CHR$(27) CHR$(36) CHR$(0)

Turning downloadable characters on or off during a line will affect *all* characters on that line. In other words, if you turn on downloadable characters anywhere in a line, all of the characters on that line will be treated as downloadable characters.

## MACRO-COMMANDS

"Macro-Instruction" is the ability of a printer to accept a single control code to do the work of a multi-command series of codes.

The Gemini's macro command capability allows you to group printer commands together and execute them with a single command. Now, this might not appear very useful at first glance, but some of you just might be scheming on ways to tame a wayward spreadsheet program or to get some cooperation from an uncooperative word processor.

First, we have to define the macro instruction.

CHR$(27) CHR$(43).....CHR$(30)

All of the commands you wish to group together in a macro replace the ..... 's. You can use up to 16 characters to define your macro. For an example, we will define a macro to set our printer to emphasized and double-width print modes. First, we look these commands up in Appendix M. There we find that the codes to enable emphasized print are

ESC E

and to define double-width

ESC W 1

After we have defined the macro, we only have to issue the select macro code to execute all the instructions in the macro. This code is

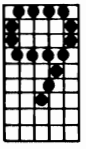
CHR$(27) CHR$(33)

Now we'll try an example and then you're free to be creative on your own.

```
10 REM Define escape character
20 ESC$=CHR$(27)
30 REM Define macro to be emphasized and expanded text
40 LPRINT ESC$;CHR$(43);ESC$;"E";ESC$;"W";CHR$(1);CHR$(30);
50 REM issue macro command and try printing
60 LPRINT ESC$;CHR$(33);
70 LPRINT "Emphasized and Expanded by Macro"
```

```
Emphasized and Expanded
by Macro
```

*Figure 8-7. Program listing using macro instructions.*



# THE FUNCTION CODES

This chapter explains the purpose, appearance, and use of the functional command codes. After you've studied this chapter, you should be able to make full use of the quick reference chart provided on the inside back cover of this manual. In addition, Appendix M provides a listing of the codes, in the same order as the reference chart, with a short description and decimal (10-base) and hexidecimal (16-base) equivalents.

The format of these descriptions is as follows:

| | |
|---|---|
| PURPOSE: | Tells what the function code accomplishes. |
| FORMAT: | Shows the correct format for the control sequence, giving both the alphanumeric and ASCII code sequences. |
| REMARKS: | Details how the command is used. |
| EXAMPLE: | Shows BASIC program segment to demonstrate the use of the command. |

## COMMANDS USED TO CONTROL PRINT STYLE

These commands are used to control the font style, the print pitch, print mode, and special effects.

### Font Style Controls

| | |
|---|---|
| PURPOSE: | ***Select the standard character set.*** |
| FORMAT: | ESC "5"<br>CHR$(27) CHR$(53) |
| REMARKS: | This command causes the printer to cancel the italic character set and select the standard character set. |
| EXAMPLE: | LPRINT CHR$(27);CHR$(53); |

PURPOSE: ***Select the italic character set.***

FORMAT: ESC "4"

CHR$(27) CHR$(52)

REMARKS: This command causes the printer to select the italics character set.

EXAMPLE:
```
LPRINT CHR$(27);CHR$(52);
```

PURPOSE: ***Select the international character set.***

FORMAT: ESC 7 *n*

CHR$(27) CHR$(55) CHR$(*n*)

REMARKS: This command causes the printer to select the international character set as shown below:

| | |
|---|---|
| *n* = 0: USA | *n* = 1: England |
| *n* = 2: Germany | *n* = 3: Denmark |
| *n* = 4: France | *n* = 5: Sweden |
| *n* = 6: Italy | *n* = 7: Spain |

EXAMPLE:
```
LPRINT CHR$(27); CHR$(55); CHR$(4)
```
. . . selects the international character set for France.

## Font Pitch Controls

PURPOSE: ***Set the print pitch.***

FORMAT: ESC "B" *n*

CHR$(27) CHR$(66) CHR$(*n*)

REMARKS: After this command is sent to the printer, subsequent printing is done: in pica (10 characters per inch) if *n* = 1; in elite (12 characters per inch) if *n* = 2; in condensed (17 characters per inch) if *n* = 3.

EXAMPLE:
```
FOR N = 1 TO 3:
   LPRINT CHR$(27);"B"; CHR$(N);
FOR I = 32 TO 112: LPRINT CHR$(I);: NEXT I
LPRINT: NEXT N
```
. . . prints out a selection of print in each of the three print pitches.

PURPOSE: ***Set the print pitch to condensed print.***

FORMAT: SI

CHR$(15)

REMARKS: A single-character command to cause the printer to shift into condensed print mode. This command is cancelled by the transmission of the DC2 code CHR$(18).

PURPOSE: ***Cause the printer to print in double-width mode.***

FORMAT: SO or ESC "W" 1

CHR$(14) or CHR$(27) CHR$(87) CHR$(1)

REMARKS: After this code is sent, subsequent data are converted to 5 CPI, 12 CPI pitch is converted to 6 CPI, and 17 CPI is converted to 8.5 CPI. The SO command is cancelled by the transmission of the DC4 code CHR$(20) or by the CR code CHR$(13). The ESC "W" 0 code will also cancel this command, causing the printer to print in normal mode.

EXAMPLE:
```
LPRINT CHR$(14)
LPRINT "Double-Width Mode"
LPRINT "Standard-Width Mode"
```
. . . prints the first line in double-width mode and the second line in standard width since the carriage return transmitted by the first LPRINT cancels the double-width mode.

PURPOSE ***Set the print pitch to condensed print mode.***

FORMAT: ESC SI

CHR$(27) CHR$(15)

REMARKS: Same as the SI code. Please refer to the SI code.

PURPOSE: ***Cause the printer to print in enlarged mode.***

FORMAT: ESC SO

CHR$(27) CHR$(14)

REMARKS: Same as the SO code. Please refer to the SO code.

PURPOSE: ***Set the print pitch to pica-sized mode.***

FORMAT: DC2

CHR$(18)

REMARKS: A single-character command to cause the printer to shift into pica-sized mode. This command is cancelled by the transmission of the ESC B 2 code, or ESC B 3, or SI code.

PURPOSE: ***Cause the printer to print in normal mode.***

FORMAT: DC4

CHR$(20)

REMARKS: A single-character command to cancel the SO code.

## Special Print Modes

PURPOSE: ***Print characters in double-strike printing mode.***

FORMAT: ESC "G"

CHR$(27) CHR$(71)

REMARKS: When the ESC "G" code is transmitted to the printer, subsequent characters are printed in the double-strike mode. The double strike is cancelled when the ESC "H" code is sent to the printer — CHR$(27) CHR$(72). When in double strike, the characters are printed once, then the paper is moved up 1/144 inch and the characters are reprinted. Shifting in and out of double-strike on the same line will cause the line to slant slightly. The double-printing mode can be used with any other print-style commands.

EXAMPLE:
```
LPRINT CHR$(27);"G";
  "This is double-strike print"
LPRINT CHR$(27);'H";
  "This is standard print"
```

PURPOSE: ***Print in emphasized printing mode.***

FORMAT: ESC "E"

CHR$(27) CHR$(69)

REMARKS: When the ESC "E" code is sent to Gemini, all subsequent characters are printed in the emphasized print mode. The emphasized print code is cancelled by sending an ESC "F" code to the printer — CHR$(27) CHR$(70). Emphasized printing can only be used with the normal pica and double-width pica characters (10 CPI and 5 CPI) and cannot be used with sub/superscripts. The emphasized print mode can be combined with double-strike mode.

## Special Printing Effects

PURPOSE: ***Print characters with underline.***

FORMAT: ESC "-" 1

CHR$(27) CHR$(45) CHR$(1)

REMARKS: When the ESC "-" 1 code is sent to the printer, all subsequent characters are printed with an underline. The underline mode is cancelled by the ESC "-" 0 code — CHR$(27) CHR$(45) CHR$(0).

PURPOSE: ***Print in superscript mode.***

FORMAT: ESC "S" 0

CHR$(27) CHR$(83) CHR$(0)

REMARKS: When the ESC "S" 0 command is received by the printer, the subsequent characters are printed in superscript mode. While in superscript, the normal bidirectional print mode is cancelled and the unidirectional print mode is invoked. Superscripts may be used with the italic font and in pica, elite, and compressed pitches. They may not be used in double-width print mode or with emphasized printing mode. Superscripts are always printed in double-strike. Superscript mode is cancelled by sending the ESC "T" code CHR$(27) CHR$(84). This code also cancels the unidirectional print mode, but does not cancel the double-strike printing mode.

PURPOSE: ***Print in subscript mode.***

FORMAT: ESC "S" 1

CHR$(27) CHR$(83) CHR$(1)

REMARKS: When the ESC "S" 1 command is received, subsequent characters are printed in the subscript style. See the remarks for superscript mode for additional information.

EXAMPLE:
```
FOR I = 48 TO 122
LPRINT CHR$(I);CHR$(27);"S";CHR$(0);
   CHR$(I);CHR$(27);
   "T";CHR$(27);"S";CHR$(1);CHR$(I);
   CHR$(27);"T"
NEXT I
```

. . . prints the upper and lowercase characters in normal superscript and subscript style.

PURPOSE: ***Print in unidirectional printing mode.***

FORMAT: ESC "U" 1

CHR$(27) CHR$(85) CHR$(1)

REMARKS: When the ESC "U" 1 code is sent to the printer, all subsequent lines are printed in unidirectional printing mode. This mode is invoked by the super/subscript mode and by the graphics mode. Unidirectional printing is useful in printing tables and charts and ensures that columnar numbers and other characters are in alignment. The ESC "U" 0 code is sent to the printer to cancel unidirectional printing and return to the faster (standard) bidirectional printing mode.

EXAMPLE:
```
LPRINT "Bidirectional printing"
FOR I = 1 TO 10: LPRINT TAB(25);"$1.29 I":
   NEXT I
LPRINT "Unidirectional printing";
   CHR$(27);"U";CHR$(1)
FOR I = 1 TO 10:
   LPRINT TAB(25);"$1.29 I": NEXT I
LPRINT CHR$(27);"U";CHR$(0);
```

## COMMANDS USED TO CONTROL VERTICAL POSITION OF THE PRINT HEAD

These commands are used to control or move the paper relative to the location of the print head. Moving the paper "up" has the effect of moving the print head "down" the page.

### Line Feed Controls

PURPOSE: ***Advance the paper by one line (line-feed command).***

FORMAT: LF

CHR$(10)

REMARKS: The line feed is set to 1/6 inch if DIP switch 1-5 is "on" upon "power up" or printer reset. If DIP switch 1-5 is "off" the line feed will be 1/8 inch. The size of the line feed can be changed using the codes listed below. When DIP switch 2-4 is "on," a line feed is automatically generated when the printer receives a CR code CHR$(13).

PURPOSE: ***Change the line feed to a pre-set value.***

FORMAT: ESC "*n*"

CHR$(27) CHR$(*m*)

REMARKS: When this command is sent to the printer, the size of the line feed is changed according to the value of "*n*" (CHR$(*m*)).

For "*n*" = "0" (*m* = 48),
successive line feeds become 1/8.
For "*n*" = "1" (*m* = 49),
successive line feeds become 7/72.
For "*n*" = "2" (*m* = 50),
successive line feeds become 1/6.

EXAMPLE: LPRINT CHR$(27);"0"

. . . changes the line-feed size to 1/8 inch, then causes a carriage return and line-feed commands to be sent to the printer.

PURPOSE: ***Change the line feed to n/72 inches.***

FORMAT: ESC "A" *n*

CHR$(27) CHR$(65) CHR$(*n*)

REMARKS: When this command is transmitted to the printer, successive line-feed commands cause the printer to advance the paper *n*/72 inch. The value of *n* must be between 1 and 127.

EXAMPLE:
```
LPRINT CHR$(27);"U";CHR$(1);
   CHR$(27);"A";CHR$(6);
FOR J = 1 TO 66: FOR I = 1 TO 80
LPRINT CHR$(239);: NEXT I:
   LPRINT: NEXT J
```

. . . causes the printer to fill a page with dots by using the special block-graphics character and setting the line feed to 1/12 inch.

PURPOSE: ***Change the line feed to n/144 inches.***

FORMAT: ESC "3" *n*

CHR$(27) CHR$(51) CHR$(*n*)

REMARKS: When this command is sent to the printer, successive line feeds are set to *n*/144 inches. The value of *n* must be between 1 and 127.

PURPOSE: ***Send a one-time-only line feed of n/144 inches.***

FORMAT: ESC "J" *n*

CHR$(27) CHR$(74) CHR$(*n*)

REMARKS: When this code is sent to the printer, it causes the paper to be advanced by *n*/144 inches. The value of *n* must be between 1 and 127. This command does not change the size of the existing line-feed value.

## Form Feed Controls

PURPOSE: ***Advance the paper to the next top of form (TOF).***

FORMAT: FF

CHR$(12)

REMARKS: When this command is received by the printer, it causes the paper to be advanced to the top of the next page or form. The initial top of form is set upon power "on." The default value of the page is set to 66 lines, but can be changed using the ESC "C" *n* and the ESC "C" 0 *n* code.

PURPOSE: ***Change the setting of the page (form) to n lines.***

FORMAT: ESC "C" *n*

CHR$(27) CHR$(67) CHR$(*n*)

REMARKS: When this code is received, the length of the page is set to *n* lines. The value of *n* should be between 1 and 127.

PURPOSE: ***Change the setting of the form length to n inches.***

FORMAT: ESC "C" 0 *n*

CHR$(27) CHR$(67) CHR$(0) CHR$(*n*)

REMARKS: When this code is received, the length of the page is set to *n* inches. The values of *n* must be between 1 and 32.

PURPOSE: ***Change the line location of the header (first line) of the form.***

FORMAT: ESC "R" *n*

CHR$(27) CHR$(82) CHR$(*n*)

REMARKS: This command is used to change the location of the first line on the next page or form. The location of the first line is given by the value of *n*, and must be between 1 and 16. The default value set upon "power up" or printer reset is *n* = 1.

PURPOSE: ***Set and enable the skip-over perforation feature.***

FORMAT: ESC "N" *n*

CHR$(27) CHR$(78) CHR$(*n*)

REMARKS: When this command is received, the skip-over perforation feature is activated. This feature causes the printer to skip to the next page or form, when the number of lines left on the page is equal to *n*. If *n* = 0, the skip-over form feature is deactivated. The value of *n* should be between 0 and 255.

EXAMPLE:
```
LPRINT CHR$(27);"C";CHR$(88);
  CHR$(27);"0";
LPRINT CHR$(27);"R";CHR$(8);
  CHR$(27);"N";CHR$(16);
```
. . . these commands allow you to print 64 lines per page. The first line starts 1 inch from the top, and the last line ends 2 inches from the bottom of the page. The line spacing is 1/8 inch.

PURPOSE: ***Cancel the skip-over perforation feature.***

FORMAT: ESC "O"

CHR$(27) CHR$(79)

REMARKS: When this command is received, the skip-over perforation set by "ESC N *n*" code and the line location of the header of the form set by "ESC R *n*" code are cancelled.

## Vertical Tabs

PURPOSE: ***Advance paper to the next vertical tab.***

FORMAT: VT

CHR$(11)

REMARKS: When this command is received, the paper is advanced to the next vertical tab position. The vertical tab positions are set upon "power up" or printer reset to lines 6, 12, 18, 24, 30, 36, 42, 48, 54, and 60. The vertical tab positions can be changed using the ESP "P" *nn* . . . 0 code.

PURPOSE: ***Change the position of the vertical tabs.***

FORMAT: ESP "P" *n*1 *n*2 *n*3 . . . 0

CHR$(27) CHR$(80) CHR$(*n*1) CHR$(*n*2) CHR$(*n*3) . . . CHR$(0)

REMARKS: When the ESC "P"...code is sent to the printer, the existing vertical tab positions are cancelled. The new vertical tabs are set to line *n*1, *n*2, *n*3, etc. The CHR$(0) character is used to end the command. The value of *n* should be between 1 and 255, and *n*1 should be less than *n*2, *n*2 less than *n*3, etc.

EXAMPLE:
```
LPRINT CHR$(13);CHR$(27);"P";CHR$(15);
  CHR$(30);CHR$(0);
FOR I = 1 TO 2: LPRINT CHR$(11);
  I: NEXT I
```
...advances the paper to the top of form and prints the numeral 1 on line 15 and the numeral 2 on line 30.

PURPOSE: ***Send a one-time-only vertical line feed of n lines.***

FORMAT: ESC"a"*n*

CHR$(27) CHR$(97) CHR$(*n*)

REMARKS: When this command is sent, it causes the paper to be advanced *n* lines vertical feed. Value of *n* must be between 1 and 127. This command does not change the present vertical tab.

EXAMPLE:
```
LPRINT CHR$(27);CHR$(97);CHR$(15)
```
...causes the paper to advance 15 vertical lines.

## COMMANDS TO CONTROL THE HORIZONTAL POSITION OF THE PRINT HEAD

These commands are used to control the horizontal position of the print head.

PURPOSE: ***Return the print head (carriage) to the next print position.***

FORMAT: CR

CHR$(13)

REMARKS: When this command is sent, the print head moves to the next print position. If DIP switch 2-2 is set to "off," this carriage return command also causes the printer to empty the print buffer. If Dip switch 2-4 is set to "on," the command causes a line-feed command to be inserted after the carriage-return code.

PURPOSE: ***Set the left-hand print margin.***

FORMAT: ESC "M" *n*

CHR$(27) CHR$(77) CHR$(*n*)

REMARKS: After this command is sent to the printer, successive carriage return codes will cause the print head to return to the *n* print position. The value of *n* should be between 1 and 255. The maximum number of print positions on an 8-inch-wide form is 80 for 10 CPI characters, 96 for 12 CPI, and 136 for 17 CPI. For the Gemini-15X the maximum number of print positions is 136 for 10 CPI, 163 for 12 CPI, and 233 for 17 CPI.

PURPOSE: ***Set the right-hand print margin.***

FORMAT: ESC "Q" *n*

CHR$(27) CHR$(81) CHR$(*n*)

REMARKS: After this command is sent to the printer, attempting to print beyond the *n* print position will cause the printer to automatically perform a line feed and return before printing the remainder of the line. The value of *n* should be between 1 and 255.

PURPOSE: ***Move the print head to the next horizontal tab location.***

FORMAT: HT

CHR$(9)

REMARKS: When the HT command is received, the print head is moved to the next horizontal tab

position. Upon power "on," the horizontal tab locations are set to columns 10, 20, 30, etc.

PURPOSE: ***Set the horizontal tab positions.***

FORMAT: ESC "D" *n*1, *n*2, *n*3. . .0

CHR$(27) CHR$(68) CHR$(*n*1) CHR$(*n*2). . . CHR$(0)

REMARKS: When the ESC "D". . . command is sent to the printer, the existing horizontal tab positions are cancelled. The new horizontal tab positions are set to print column *n*1, *n*2, *n*3,. . . The maximum number of horizontal tabs is 255. The CHR$(0) code must be at the end of this command. The value of *n*1 must be less than *n*2, *n*2 must be less than *n*3, etc.

EXAMPLE:
```
LPRINT CHR$(27);"M";CHR$(5);CHR$(27);
  "Q";CHR$(70);
LPRINT CHR$(27);"D";CHR$(10);CHR$(30);
  CHR$(0)
```

. . . sets the left-hand margin at print column 5, sets the right-hand margin at print column 70, and sets the two horizontal tab positions: one at print column 10, and the other at print column 30.

PURPOSE: ***Move the print head back one space.***

FORMAT: BS

CHR$(8)

REMARKS: When the BS code is sent to the printer, the print head is shifted one print column to the left. The BS code can be used to overprint characters.

PURPOSE: ***Send a one-time-only horizontal tab of n columns.***

FORMAT: ESC b *n*

CHR$(27) CHR$(98) CHR$(*n*)

REMARKS: This command enables the print head to shift *n* columns to the right. The value must be between 1 and 127. This does not change the present horizontal tab.

EXAMPLE: LPRINT CHR$(27);CHR$(98);CHR$(45)

. . . shifts the print head 45 columns to the right.

## COMMANDS USED TO CONTROL GRAPHICS

PURPOSE: ***Print low-resolution (normal-density) graphics.***

FORMAT: ESC "K" *n*1 *n*2 *m*1 *m*2. . .

CHR$(27) CHR$(75) CHR$(*n*1) CHR$(*n*2) CHR$(*m*1) CHR$(*m*2). . .

REMARKS: This command selects the 60-dots-per-inch, column-scan, bit-image graphics mode. The number of columns to be printed is given by *n*1 + 256 * *n*2. There must be *n*1 + 256 * *n*2 characters following *n*2. These characters determine which pins are fired according to the explanation found in Chapter 6 for dot graphics.

PURPOSE: ***Print high-resolution (double-density) graphics.***

FORMAT: ESC "L" *n*1 *n*2 *m*1 *m*2. . .

CHR$(27) CHR$(76) CHR$(*n*1) CHR$(*n*2) CHR$(*m*1) CHR$(*m*2). . .

REMARKS: This command selects the 120-dots-per-inch graphics mode. See remarks for the ESC "K". . .code for additional details.

PURPOSE: ***Print high-resolution (double-density) graphics with double speed.***

FORMAT: ESC "y" *n*1 *n*2 m1 m2. . .

CHR$(27) CHR$(121) CHR$(*n*1) CHR$(*n*2) CHR$(*m*1) CHR$(*m*2). . .

REMARKS: This command selects the 120-dots-per-inch graphics mode but with double the print speed.
See ESC "K"...code for additional details.

PURPOSE: ***Print quadruple-density graphics.***

FORMAT: ESC "z" *n*1 *n*2 *m*1 *m*2...
CHR$(27) CHR$(122) CHR$(*n*1) CHR$(*n*2) CHR$(*m*1) CHR$(*m*2)...

REMARKS: This command selects the 240-dots-per-inch column scan bit image graphics mode.
See ESC "K"...code for additional details.

## COMMANDS USED FOR MACRO INSTRUCTION

These commands are used to control the macro instruction.

PURPOSE: ***Define the macro instruction.***

FORMAT: ESC"+"...RS
CHR$(27) CHR$(43)...CHR$(30)

REMARKS: When this code is sent to the printer, the existing macro instruction is cancelled. The new macro instruction is set. The maximum number of instruction bytes is 16. The RS(CHR$(30)) character is used to end the command.

PURPOSE: ***Select the macro instruction.***

FORMAT: ESC "!"
CHR$(27) CHR$(33)

REMARKS: When this code is sent to the printer, the macro instruction set by "ESC +...RS" code is selected.

## COMMANDS USED TO DOWNLOAD CHARACTERS

These commands are used to control the download characters. For details, please refer to Chapter 8.

PURPOSE: ***Define the download character into RAM.***

FORMAT: ESC "*" 1 *n*1 *n*2 *m*1 *m*2 . . .

CHR$(27) CHR$(42) CHR$(1) CHR$(*n*1) CHR$(*n*2) CHR$(*m*1) CHR$(*m*2). . .

REMARKS: When you define the download character, character patterns (*m*1 *m*2. . .) are inputted with 9 bytes into RAM after transmitting the "ESC * 1" code, "*n*1" as designating position to be written, and "*n*2" as the descender data. The value of *n*1 must be between 32 and 126. The value of *n*2 must be "1" when the character is to be shifted two dots downward, and "0" when it is not shifted.

PURPOSE: ***Copy the fonts in character ROM into download RAM.***

FORMAT: ESC "*" 0

CHR$(27) CHR$(42) CHR$(0)

REMARKS: When this command is received, the fonts in character ROM are copied into the download character RAM.

PURPOSE: ***Select the download character set.***

FORMAT: ESC "$" 1

CHR$(27) CHR$(36) CHR$(1)

REMARKS: This command causes the printer to select the download character set. Download character sets cannot be intermixed with other character sets on the same line.

PURPOSE: ***Cancel the download character set.***

FORMAT: ESC "$" 0

CHR$(27) CHR$(36) CHR$(0)

REMARKS: This command causes the printer to cancel the download character set and select the standard ASCII character set.

## OTHER FUNCTION CODES

PURPOSE: ***Control the value of the eight bits of data sent to printer.***

FORMAT: ESC ">" or ESC "=" or ESC "#"

CHR$(27) CHR$(62) or CHR$(27) CHR$(61) or CHR$(27) CHR$(35)

REMARKS: These codes are used to control the value of the eighth bit of the data that is sent to the printer. After the ESC ">" code is received by the printer, all subsequent data is processed with the eighth regarded as set to 1. The ESC "#" code is used to cancel the ESC ">" command. After the ESC "=" code is received, all subsequent data is processed with the eighth bit regarded as set to 0. The ESC "#" code is used to cancel the ESC "=" command. These codes are provided to allow users with a 7-bit interface to access the special characters whose ASCII code is greater than 127. These codes should be used to transmit printer function codes. ESC ">" only functions with the DIP switch number 2-3 set to "on," for a 7-bit interface.

PURPOSE: ***Delete the last character sent to the printer.***

FORMAT: DEL

CHR$(127)

REMARKS: When the CHR$(127) code is sent to the printer, it causes the printer to delete the last character it received, with the exception of function codes.

PURPOSE: ***Put the printer in "online" mode.***

FORMAT: DC1

CHR$(17)

REMARKS: This code cancels the DC3 code and puts the printer back in the "online" mode.

PURPOSE: ***Put printer in "offline" mode.***

FORMAT: DC3

CHR$(19)

REMARKS: When this code is sent to the printer, Gemini assumes the "offline" mode. All subsequent characters and control codes sent to the printer are disregarded. The only control code that is processed is CHR$(17)(DC1), which cancels CHR$(19), and puts the printer back "on line."

PURPOSE: ***Activate the Gemini "buzzer."***

FORMAT: BEL

CHR$(7)

REMARKS: When the CHR$(7) code is sent to Gemini, the result is that the built-in buzzer "beeps" for one-fourth of a second. The buzzer function can be temporarily deactivated by sending an ESC "Y" 0 (CHR$(27) CHR$(89) CHR$(0) control code) to the printer. After the ESC "Y" 0 code is received, the CHR$(7) no longer causes the buzzer to "beep." The buzzer can be reactivated by sending an ESC "Y" 1 (CHR$(27) CHR$(89) CHR$(1) code) to the printer.

PURPOSE: ***Allow printing even though the "paper-out" detector indicates Gemini is out of paper.***

FORMAT: ESC "8"

CHR$(27) CHR$(56)

REMARKS: When the ESC "8" function code is sent to Gemini, the signal from the "paper-out" detector is disregarded. This code allows you to print on a single sheet. If the "paper-out" signal were not disregarded, Gemini would stop printing part way down the single sheet because the printer would "think" it's out of paper. The "paper-out" signal can be restored by sending an ESC "9" function code (CHR$(27) CHR$(57)).

PURPOSE: ***Regard the signal from the "paper-out" detector.***

FORMAT: ESC "9"

CHR$(27) CHR$(57)

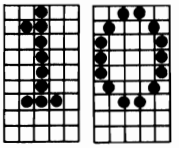
REMARKS: When this code is sent to the printer, the signal from the "paper-out" detector is restored.

PURPOSE: ***Initialize the printer conditions.***

FORMAT: ESC "@"

CHR$(27) CHR$(64)

REMARKS: When this code is sent to the printer, the conditions of the printer are initialized.



# MAINTENANCE

The purpose of "maintenance" is to maximize the life of your Gemini. Toward that end, the best maintenance is "preventive."

The Gemini printer is a well-engineered peripheral, and it serves as a valuable information-handling tool when used correctly. Part of using Gemini correctly is paying attention to the operating environment. Chapter 1 outlines the suggested considerations for keeping that environment clean.

Periodic cleaning of the exterior and interior of your printer will help keep Gemini going for years. How often you need to clean depends on how "dirty" the operating environment is.

## CLEANING GEMINI

Exterior cleaning: When dust, grease, or other material accumulates on the outside of your printer, wipe the external surfaces of the case carefully with a clean cloth moistened with alcohol.

***Note:*** Exercise the utmost care not to wet the electronic parts and machine body.

Interior cleaning: Remove dust and dirt from the internal mechanism by gently using a soft brush. Be very careful not to damage the electronic parts or wiring circuit.

To clean the interior of Gemini, you should remove the upper casing, as described below.

## REMOVING THE UPPER CASING

***Warning:*** Unplug your Gemini before removing the upper casing.

When you need to set Dip Switch number 1 or change other parts of the printer, the upper casing must first be removed. Figure 10-1 shows the position of the three screws that hold the upper casing in place on the Gemini-15X. Figure 10-2 shows the location of the two screws that do the same for the Gemini 10-X. In both cases, the mounting screws are Phillips-head types.

To replace the fuse, turn the power switch "off." Just to be on the safe side, make sure you unplug the power cord from the power source. Then remove the platen knob, take out the mounting screws, and remove the upper casing. To remove the upper casing, place your fingers on the inside front of the printer with your thumbs pressing on the outside front of the lower casing. Gently pull the upper-casing front toward you until it pops free. Set the upper-casing front on end (next to the printer) so that the short wires attached to the control panel remain connected.

After removing the casing, check the fuse, which is located on the fuse board near the power switch. The fuse is a commonly used type, with a metal strip suspended in a glass and metal case. If the strip is broken, the fuse is "blown."

If the fuse is blown, replace it with a BELL's 5MT1.25 (1.25A, 125V) "slo-blo" type fuse, or equivalent. Just slip the old fuse out and snap the replacement fuse into place in the two clasps (as shown in Figure 10-3).

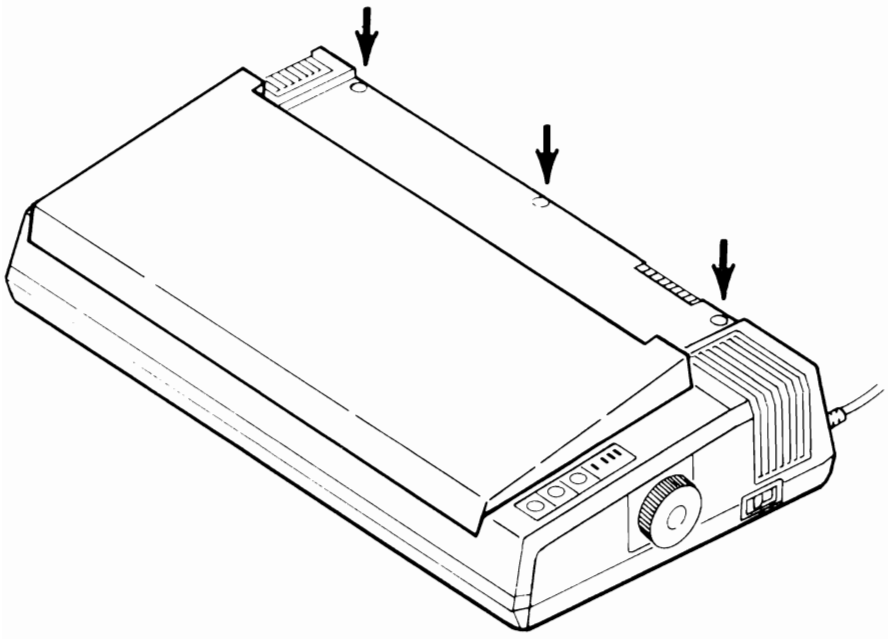


*Figure 10-1. Three mounting screws hold the Gemini-15X upper casing in place.*

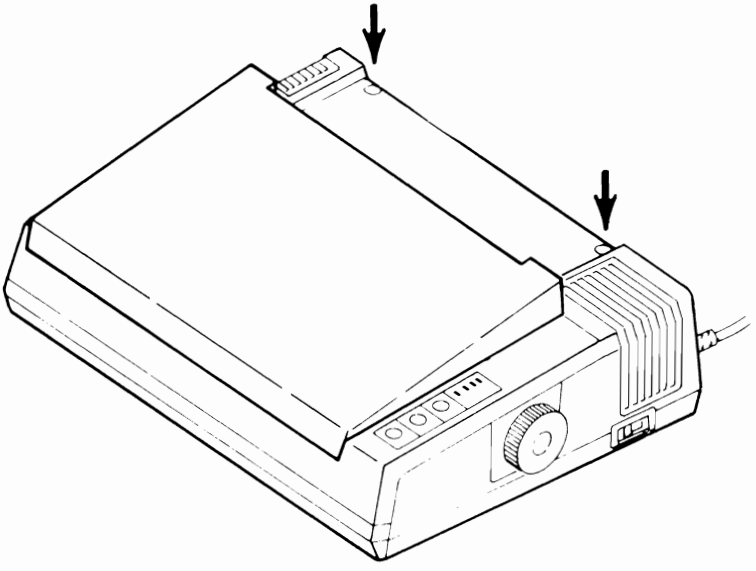


*Figure 10-2. Two mounting screws hold the upper casing in place on the Gemini-10X.*

After replacing (or just checking) the fuse, re-attach the upper casing (reversing the procedure described above) and insert the mounting screws and the platen knob. (Refer to Figure 1-6 in Chapter 1 for an "exploded view" of Gemini-10X with its upper casing removed.)

You probably need to replace the fuse when all of the following four statements apply to your situation:

- the printer won't operate,
- the power-on lamp isn't lit on the control panel,
- you are sure that the power switch is "on", and
- the printer is connected to a power source.

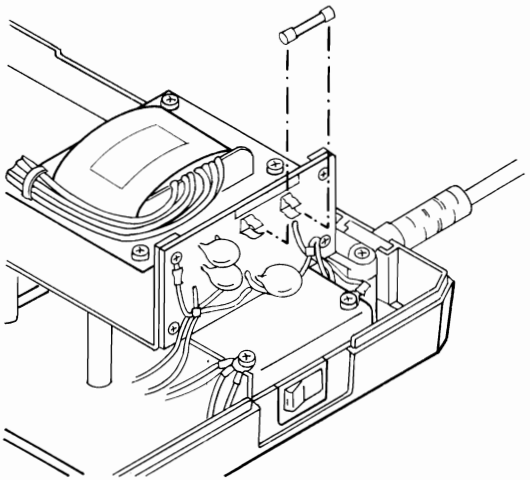


*Figure 10-3. Replacing a fuse is simple, but make sure you use the correct type: BELL's 5MT1.25 1.25A, 125V or an equivalent fuse.*

## REPLACING THE PRINT HEAD

When you replace the print head, start by removing the ink ribbon. You might want to refer to Figure 10-4 as you follow these steps:

***Warning:*** The print head will be hot during operation and soon afterward, so be careful not to burn yourself.

1. Extract the head cable from the head connector. (Hold down the head cable board while performing this disconnection.)

2. Remove the two clamping screws from the print head.

3. Detach the print head.

4. Place the new print head in position, and attach it with the clamping screws. Apply "screw lock" (an adhesive) to the heads of the screws.

5. Insert the head cable tightly into the head connector. (Be sure the connection is a good one, otherwise it may cause problems.)

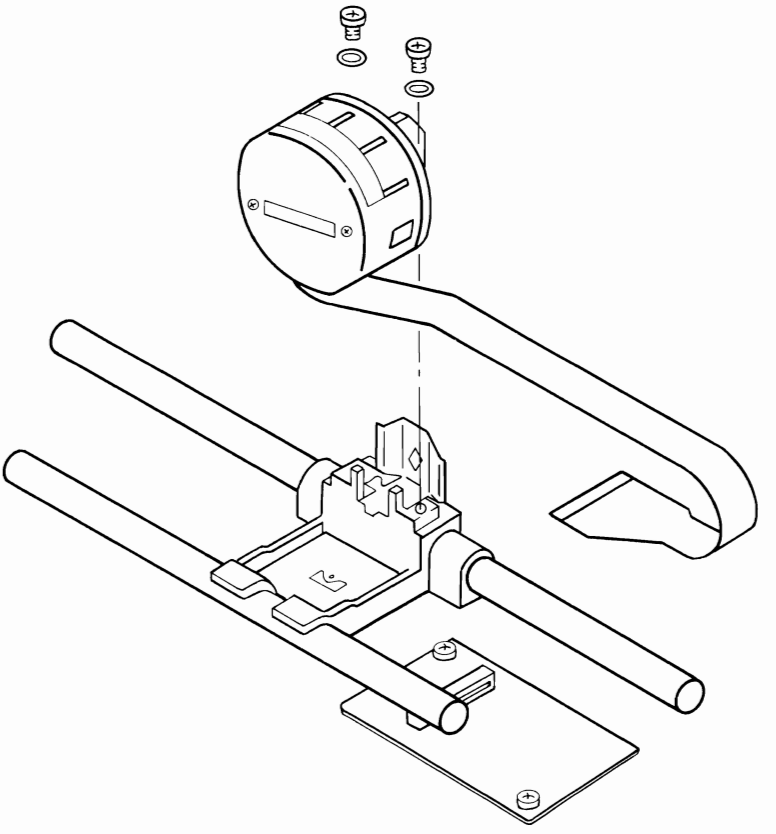


*Figure 10-4. Replacing the print head by yourself is easy, quick, and avoids downtime.*

## PERIODIC INSPECTIONS

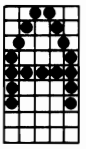
To keep Gemini in tip-top shape, you should make the following inspections every six months (under normal use). Expressed another way, you should "check under the hood" every half-million lines of printing. Here's what to check:

**Maintenance Check List**

- ☐ Clean the print head and sensor peripherals with a soft brush.
- ☐ Check the various screws and make sure they are tight. If they aren't, tighten them with a screwdriver.
- ☐ Rotate the platen, and check for possible damage. If the platen is worn or uneven, replace it.
- ☐ Remain alert to "abnormal" operating sounds. Strange noises can signal strange conditions, and the earlier you recognize and fix a problem, the better.
- ☐ Perform Self-Testing and check the movement of each mechanism.

# Notes

appleII
commodore
ATARI
IBM
OSBORNE
TRS 80



# APPLE II PLUS / IIE

## The Interface Board

You need to acquire and install an Apple interface board (also called a "card") and a connecting cable. Your Gemini dealer can answer any questions you have about the board or its installation in slot number 1. Star Micronics, Inc. (the makers of Gemini), has created the **grafstar™** intelligent interface especially for your Apple II Plus or Apple IIe computers.

## The Connecting Cable

Use the Apple, Centronics compatible, parallel I/F cable, modified to ground the 8th bit, or use a cable supplied by your Gemini dealer.

## Setting the DIP Switches

Because Apple uses a seven-bit interface, set DIP switch number 2-3 to the "on" position. Because BASIC supplies a line feed at the end of each line, turn DIP switch number 2-4 to the "off" position (unless you want to generate "double-spaced" output, in which case turn number 2-4 "on"). Switch number 2-1 can be set to either position.

Switch number 2-2 should be set to "off" if you want Gemini to print every time it receives a CR code; set number 2-2 to "on" if you wish to have the carriage return ignored and to print only when the buffer is full.

## The Use of Basic

In this manual, we used the BASIC programming language to illustrate how to control the codes that manipulate Gemini's operations. In our sample programs, the CHR$ function is used to send control codes to the printer. Unfortunately, this function is not supported by Apple Integer BASIC. However, with the aid of the follow-

ing notes, you should be able to run all the sample programs presented in this manual if you use Applesoft BASIC.

Applesoft BASIC uses the PRINT command to send information to both the screen and the printer.

The PR # 1 command sends the information only to the printer (when the printer interface is in slot number 1), and the PR # 0 command sends the information only to the screen.

Therefore, you should use:

```
PRINT   CHR$(4)   "PR # 1"
PRINT
```

instead of the LPRINT command. In addition, use:

```
PRINT   CHR$(4)   "PR # 0"
PRINT
```

instead of the PRINT command.

The Apple video display is 40 columns wide. If you want the print mode to print more than 40 columns, you need to send an additional command code:

```
PRINT   CHR$(4)   "PR # 1"
PRINT   CHR$(9)   "255N"
PRINT
```

This message allows Gemini to print a line of any length. This is particularly important when sending bit-image graphics to Gemini.

The CHR$(9) (or function code HT, which moves the print head to the next tab position) and CHR$(13) (or function code CR for a carriage return) codes cause problems when used in both the Gemini escape-function code sequences and in the graphics mode. Avoid using these codes if possible.

The printer interface card does not allow us to send ASCII codes greater than 127. If you try to send these codes, the interface sets the eighth bit to zero. The result, for example, is that PRINT CHR$(239) would be received by the printer as PRINT CHR$(111); that is, 239 - 128 = 111. Gemini provides the ESC ">" function code to allow 7-bit interface users to overcome this problem. See Chapter 9 for a discussion of this problem.

Apple users (and other 7-bit interface users) should also be aware of the difference between 7-bit and 8-bit interface graphics. See Chapter 6 for a discussion on this topic.

## Pin Configurations

The following chart shows the pin-outs for Gemini-10X and Gemini-15X printers and the Apple II Plus, using a parallel interface board.

| Gemini-10X/15X | | Parallel Apple Board | |
|---|---|---|---|
| 25. | SG | 1. | SG |
| 26. | SG | 2. | SG |
| 27. | SG | 3. | SG |
| 1. | $\overline{\text{STROBE}}$ | 4. | $\overline{\text{STROBE}}$ |
| 28. | SG | 5. | N/C |
| 2. | D1 | 6. | D1 |
| 3. | D2 | 7. | D2 |
| 4. | D3 | 8. | D3 |
| 5. | D4 | 9. | D4 |
| 6. | D5 | 10. | D5 |
| 7. | D6 | 11. | D6 |
| 8. | D7 | 12. | D7 |
| 9 | D8 | 13. | D8 |
| 10. | $\overline{\text{ACK}}$ | 14. | $\overline{\text{ACK}}$ |
| 29. | SG | 15. | SG |

***Note:*** If you purchased a Star Micronic's "intelligent" Apple Interface Board, it comes complete with the cable. Just plug it in, and it's ready to go!

## Program Listings

```
4  REM  ††
5  REM  †† PRINTING EFFECTS PROG. ††
6  REM  ††
10  PR# 1: PRINT  CHR$ (9) + "132N": PRINT
20 ESC$ =  CHR$ (27):LF$ =  CHR$ (10):FF$ =  CHR$ (12):CR$ =  CHR$ (13)
30 PICA$ = ESC$ + "B" +  CHR$ (1):ELITE$ = ESC$ + "B" +  CHR$ (12):COMP$ =
    ESC$ + "B" +  CHR$ (3)
40 SPR$ = ESC$ + "S" +  CHR$ (0):SUB$ = ESC$ + "S" +  CHR$ (1):CANS$ = ESC
    $ + "T"
50 ITALIC$ = ESC$ +  CHR$ (52)
55 STND$ = ESC$ +  CHR$ (53)
60 WIDE$ =  CHR$ (14):CWIDE$ =  CHR$ (20):PRSET$ = ESC$ + "@"
70 EMPH$ = ESC$ + "E":CEMPH$ = ESC$ + "F":DBLS$ = ESC$ + "G":CDBLS$ = ESC$
     + "H"
80  PRINT PRSET$;ESC$;"R"; CHR$ (15);FF$
90  FOR I = 1 TO 2
```

```
100  IF I = 1 THEN  PRINT WIDE$;"      GEMINI STANDARD TYPE STYLES";CWIDE$
     ;LF$
110  IF I = 2 THEN  PRINT ITALIC$;WIDE$;"      GEMINI ITALIC TYPE STYLES":
      PRINT CWIDE$;LF$;LF$
120  PRINT ESC$;">": FOR N = 1 TO 40: PRINT  CHR$ (231); CHR$ (232);: NEXT
     N: PRINT ESC$;"=": PRINT : PRINT
130  PRINT "          17 CPI  12 CPI  10 CPI    8.5 CPI     6 CPI
     S CPI"
140  PRINT ESC$;">": FOR N = 1 TO 80: PRINT  CHR$ (241);: NEXT N: PRINT ES
     C$;"=": PRINT
150 M1$ = "  ABC":M2$ = "X" +  CHR$ (120):M3$ = "Y" +  CHR$ (121)
160  FOR J = 1 TO 4
170  IF I = 2 THEN  PRINT ITALIC$;
180  IF J = 1 THEN  PRINT COMP$;"NORMAL          ";
190  IF J = 2 THEN  PRINT COMP$;"DOUBLE STRIKE  ";
200  IF J = 3 THEN  PRINT COMP$;"EMPHASIZED     ";
210  IF J = 4 THEN  PRINT COMP$;"DOUBLE STRIKE  ";
220  FOR K = 1 TO 6
230  IF K = 1 THEN  PRINT COMP$;
240  IF K = 2 THEN  PRINT ELITE$;
250  IF K = 3 THEN  PRINT PICA$;
260  IF K = 4 THEN  PRINT COMP$;WIDE$;
270  IF K = 5 THEN  PRINT ELITE$;WIDE$;
280  IF K = 6 THEN  PRINT PICA$;WIDE$;
290  IF J = 2 THEN  PRINT DBLS$;
300  IF J = 3 THEN  PRINT EMPH$;
310  IF J = 4 THEN  PRINT DBLS$;EMPH$;
320  PRINT M1$;SPR$;M2$;CANS$;SUB$;M3$;CANS$;CDBLS$;
330  IF K > 3 THEN  PRINT CWIDE$;
340  NEXT K
345  IF J = 2 THEN  PRINT LF$;
350  IF J = 4 THEN  PRINT ESC$;"A"; CHR$ (8);LF$;COMP$;"     + ";CR$;"EMPH
     ASIZED   ";
360  PRINT PRSET$
370  NEXT J
380  PRINT
390  NEXT I
400  PRINT FF$: PRINT  CHR$ (9) + "80N": PR# 0
410  END
```

*Figure A-1 is the font styles program (refer to Figure 5-1) amended for the Apple II computer.*

```
5 BL$ = "                                      "
7 LI$ = "--------------------------------------"
10  HOME : DIM A$(22,40)
20  HOME :X = 1:Y = 1
30  GOSUB 150: VTAB X: HTAB Y: PRINT "";
40  GET B$
50  IF B$ =  CHR$ (5) THEN  HOME : END
60  IF B$ =  CHR$ (18) THEN  HOME : GOSUB 200: GOTO 20
70  IF B$ =  CHR$ (16) THEN  GOTO 210
80  IF B$ =  CHR$ (13) THEN A$(X,Y) = B$: VTAB X: HTAB Y: PRINT "";X = X +
    1:Y = 1: VTAB X: HTAB Y: PRINT "";: GOTO 40
90  IF B$ =  CHR$ (4) THEN  GOSUB 130: GOTO 40
100  VTAB X: HTAB Y: PRINT B$;:A$(X,Y) = B$:Y = Y + 1
110  IF Y = 40 THEN A$(X,Y) =  CHR$ (13): PRINT  CHR$ (7);:X = X + 1:Y = 1
    : GOTO 40
120  VTAB X: HTAB Y: PRINT "";: GOTO 40
130  VTAB X: HTAB Y: PRINT " ";:Y = Y - 1: IF Y = 0 THEN Y = 1: PRINT  CHR$
    (7);
135 A$(X,Y) = ""
140  VTAB X: HTAB Y: PRINT "";: RETURN
150  VTAB 24: HTAB 1: INVERSE : PRINT " ^P)RINT";
160  VTAB 24: HTAB 10: PRINT "^D)ELETE";
170  VTAB 24: HTAB 19: PRINT "^R)ESTART";
180  VTAB 24: HTAB 29: PRINT "^E)XIT ";
185  NORMAL
187  VTAB 22: HTAB 1: PRINT LI$;
190  RETURN
200  VTAB 10: HTAB 10: PRINT "PLEASE STAND BY";: FOR I = 1 TO X: FOR J = 1
     TO 40:A$(I,J) = "": NEXT J: NEXT I: RETURN
210  REM
220  PR# 1: PRINT  CHR$ (27);"@": PR# 0
230  HOME : VTAB 24: HTAB 1: INVERSE : PRINT "C)OMMAND";: HTAB 10: PRINT "
    P)RINT";
240  HTAB 17: PRINT "R)ETURN";: NORMAL
250  GET G$
260  IF G$ = "P" THEN  GOSUB 310
270  IF G$ = "C" THEN  GOSUB 320: GOTO 230
280  IF G$ = "R" THEN  GOSUB 300: GOTO 30
290  GOTO 250
300  HOME : FOR I = 1 TO X: FOR J = 1 TO 40: VTAB I: HTAB J: PRINT A$(I,J)
    ;: NEXT J: NEXT I: RETURN
310  PR# 1
```

```
311  PRINT  CHR$ (9);"40N"
312  FOR I = 1 TO X: FOR J = 1 TO 40
313  PRINT A$(I,J);: NEXT J: NEXT I
314  PR# 0
315  VTAB 24: HTAB 30
317  RETURN
320  VTAB 24: HTAB 1: PRINT BL$;
330  REM
340  VTAB 24: HTAB 1: INVERSE : PRINT "E) ESC";:
350  HTAB 8: PRINT "C) CHR$";: HTAB 16: PRINT "Q) END COMMAND";
355  NORMAL
360  GET G$
370  IF G$ = "E" THEN  GOSUB 420: GOTO 400
380  IF G$ = "C" THEN  GOSUB 530: GOTO 400
390  IF G$ = "Q" THEN  GOTO 410
400  GOTO 360
410  RETURN
420 I = 0: VTAB 22: HTAB 4: PRINT "ESC ";
430  GET G$
440  IF G$ =  CHR$ (13) THEN 480
450  PRINT G$;
455  IF I > 0 THEN 470
460  PRINT " ";
470 S$(I) = G$:I = I + 1: GOTO 430
480  IF I > 2 THEN  FOR J = 1 TO I:D$ = D$ + S$(J): NEXT J:ST$ =  CHR$ (27
     ) + S$(0) +  CHR$ ( VAL (D$)):D$ = ""
490  IF I = 2 THEN ST$ =  CHR$ (27) + S$(0) +  CHR$ ( VAL (S$(1)))
500  IF I = 1 THEN ST$ =  CHR$ (27) + S$(0)
510  PR# 1: PRINT  CHR$ (9);"40N": PRINT ST$:ST$ = "": PR# 0: FOR J = 1 TO
     I:S$(J) = "": NEXT J:I = 0
520  VTAB 22: HTAB 4: PRINT BL$;: RETURN
530  VTAB 22: HTAB 2: PRINT "CHR$ ";:ST$ = ""
540  GET G$
550  IF G$ =  CHR$ (13) THEN 570
560  PRINT G$;:ST$ = ST$ + G$: GOTO 540
570  VTAB 22: HTAB 2: PRINT BL$;:
580  PR# 1: PRINT  CHR$ (9);"40N"
590  PRINT  CHR$ ( VAL (ST$));:ST$ = "": PR# 0: RETURN
```

*Figure A-2 is the word processing program (refer to Figure 5-4) amended for the Apple II computer.*

```
5  PR# 1
10  PRINT  TAB( 30);"HIGH RESOLUTION VERTICAL SPACING"
15  PRINT  TAB( 30);"USING THE ESC 3N CONTROL CODE"
20  FOR I = 1 TO 30
25  IF I = 7 THEN  GOTO 110
26  IF I = 9 THEN  GOTO 110
30  IF I = 13 THEN  GOTO 110
40 M =  INT (144 / (6 * I)): IF M < 2 THEN M = 2
50  PRINT  CHR$ (27); CHR$ (51); CHR$ (I);
60  FOR J = 0 TO M
70  IF J = M - 1 THEN  PRINT I;"/144 INCH SPACING          ";: GOTO 90
80  PRINT  TAB( 30);
90  FOR N = 1 TO 40: PRINT  CHR$ (27);">"; CHR$ (241); CHR$ (27);"=";: NEXT
    N: PRINT
100  NEXT J
110  NEXT I
120  PRINT  CHR$ (27);"@"
130  PR# 0: END
```

*Figure A-3 is the vertical spacing program (refer to Figure 6-5) amended for the Apple II computer.*

```
10  PR# 1
20  PRINT  CHR$ (27);"@"; CHR$ (27);"A"; CHR$ (6);
30  FOR LI = 1 TO 2
40  PRINT  CHR$ (27);"K"; CHR$ (114); CHR$ (0);
50  FOR SPACE = 1 TO 100
55  PRINT  CHR$ (0);: NEXT SPACE
60  FOR COLM = 1 TO 14
70  READ C: PRINT  CHR$ (C);: NEXT COLM
80  PRINT
90  NEXT LI
100  DATA   14,16,34,64,78,78,64,64,78,78,64,34,16,14
110  DATA  224,16,232,68,34,18,18,18,18,34,68,232,16,224
120  PRINT  CHR$ (27);"@"
130  PR# 0
140  END
```

*Figure A-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the Apple II computer.*

```
10  PR# 1
20  PRINT  CHR$ (27);">";
30  PRINT  CHR$ (163); CHR$ (161); CHR$ (27);"A"; CHR$ (6)
40  PRINT  CHR$ (162); CHR$ (160); CHR$ (27);"2"
50  PRINT  CHR$ (27);"="; PR# 0: END
```

*Figure A-5 is the circle program presented in chapter 7, amended for the Apple II computer.*

```
10  PR# 1
30  PRINT  CHR$ (32); CHR$ (27);">"; CHR$ (163); CHR$ (27);"A"; CHR$ (6)
40  PRINT  CHR$ (160)
50  PRINT  CHR$ (27);"="
60  PR# 0: END
```

*Figure A-6 is the tilde program presented in Chapter 7 amended for the Apple II computer.*

```
1  PR# 1
10  PRINT  TAB( 20);"N"
15  PRINT  TAB( 20); CHR$ (27);">"; CHR$ (164); CHR$ (27);"="
20  PRINT  TAB( 17);"W";
25  PRINT  CHR$ (27);">"; CHR$ (166); CHR$ (27);"=";
30  PRINT "   "; CHR$ (27);">"; CHR$ (167); CHR$ (27);"=";"E"
35  PRINT  TAB( 20); CHR$ (27);">"; CHR$ (165); CHR$ (27);"="
40  PRINT  TAB( 20);"S"
45  PR# 0: END
```

*Figure A-7 is the map program presented in Chapter 7 amended for the Apple II computer.*

```
10  REM  WY LARGE LETTERS
20  PR# 1
25  PRINT  CHR$ (27);"U"; CHR$ (1);
30  PRINT  CHR$ (27);"A"; CHR$ (6)
40  FOR J = 1 TO 9
50  FOR I = 1 TO 12
60  READ A
70  PRINT  CHR$ (27);">"; CHR$ (A);
```

```
80  NEXT I
90  PRINT
100  NEXT J
110  DATA  239,224,224,224,239,224,224,224,224,224,224,224
111  DATA  239,224,224,224,239,224,224,224,224,224,224,224
112  DATA  239,224,224,224,239,224,239,224,224,224,239,224
113  DATA  239,224,239,224,239,224,234,233,224,234,233,224
114  DATA  239,234,239,233,239,224,224,239,224,239,224,224
115  DATA  239,239,224,239,239,224,224,234,239,233,224,224
116  DATA  234,233,224,234,233,224,224,224,239,224,224,224
117  DATA  224,224,224,224,224,224,224,234,233,224,224,224
118  DATA  224,224,224,224,224,224,224,239,224,224,224,224
120  PRINT  CHR$ (27);"@"
130  PR# 0: END
```

*Figure A-8 is the "Wy" program (refer to Figure 7-1) amended for the Apple II computer.*

```
1  REM  **
2  REM  ** BOAT PROG. **
3  REM  **
10  PR# 1: PRINT  CHR$ (27);"A"; CHR$ (6); CHR$ (15)
12  PRINT  CHR$ (27);"U"; CHR$ (1);
15  FOR K = 1 TO 3
16  PRINT  CHR$ (27);"B"; CHR$ (K)
17  RESTORE
20  FOR J = 1 TO 9
30  FOR I = 1 TO 12
40  READ A
50  PRINT  CHR$ (27);">"; CHR$ (A);
60  NEXT I
70  PRINT
80  NEXT J
90  DATA  224,224,224,224,224,224,224,224,224,245,224,224
91  DATA  224,224,224,224,224,224,  224,224,252,245,224,224
92  DATA   224,224,224,245,224,224,  224,252,239,245,224,224
93  DATA  224,224,252,245,224,224,  252,239,239,245,224,224
94  DATA  224,252,239,245,224,252,  239,239,239,245,224,224
95  DATA  252,239,239,245,252,239,  239,239,239,245,224,224
```

```
96  DATA  241,241,241,249,241,241, 241,241,241,249,224,224
97  DATA  239,239,239,239,239,239, 239,239,239,239,239,251
98  DATA  253,239,239,239,239,239, 239,239,239,239,251,224
99  PRINT : NEXT K
100  PRINT  CHR$ (27);"@": PR# 0: END
```

*Figure A-9 is the boat program (refer to Figure 7-2) amended for the Apple II computer.*

```
4  HOME : PRINT "": PRINT ""
5  PRINT " ": PRINT " "
6  PRINT "THIS PROGRAM TAKES ABOUT"
7  PRINT "3 MINUTES TO RUN. PLEASE"
8  PRINT "TURN ON YOUR PRINTER AND"
9  PRINT "STAND BY....."
10 A = 24576
20  FOR I = A TO A + 12
30  READ B
35  POKE I,B
40  NEXT I
50  DATA  32,74,255,165,250,5,251
60  DATA  133,252,32,63,255,96
100  REM  MICRO-PLOT
110  DIM BIT%(76,12)
1000  REM  SET PROGRAM CONSTANTS
1010 MASK%(1) = 64:MASK%(4) = 8
1020 MASK%(2) = 32:MASK%(5) = 4
1030 MASK%(3) = 16:MASK%(6) = 2
1040 LX = 20:LY = 20
1050 XFAC = 72 / LX:YFAC = 75 / LY
2000  REM  PLOT CURVE
2010 RAD = 9
2020 X1 = 19:Y1 = 10
2030  FOR ANG = 0 TO 360 STEP 10
2040 R1 = ANG * 6.28 / 360
2050 X2 = RAD *  COS (R1) + 10:Y2 = RAD *  SIN (R1) + 10
2060  GOSUB 4000
2070  NEXT ANG
2080  FOR ANG = 0 TO 360 STEP 15
```

```
2090 R1 = ANG * 6.28 / 360
2100 R2 = (ANG + 150) * 6.28 / 360
2110 X1 = RAD *  COS (R1) + 10:Y1 = RAD *  SIN (R1) + 10
2120 X2 = RAD *  COS (R2) + 10:Y2 = RAD *  SIN (R2) + 10
2130  GOSUB 4000
2140  NEXT ANG
2150 BIT%(39,0) = 4
3000  REM  SEND BIT IMAGE MAP TO PRINTER
3005  PR# 1
3006  PRINT  CHR$ (9);"255N"
3010  PRINT  CHR$ (27);"A"; CHR$ (6)
3015  PRINT  CHR$ (13)
3020  FOR ROW = 0 TO 11
3025  PRINT  CHR$ (27);"K"; CHR$ (119); CHR$ (1);
3050  FOR COL = 1 TO 75
3055  FOR LL = 1 TO 5
3060  PRINT  CHR$ (BIT%(COL,ROW));
3065  NEXT LL
3070  NEXT COL
3080  PRINT
3090  NEXT ROW
3100  PRINT
3102  PR# 0
3110  END
4000  REM   DRAW A LINE FROM X1,Y1 TO X2,Y2
4010 XL = X2 - X1:YL = Y2 - Y1
4020 NX =  ABS (XL * XFAC):NY =  ABS (YL * YFAC)
4030  IF NX < NY THEN NX = NY
4040 NS% =  INT (NX + 1)
4050 DX = XL / NS%:DY = YL / NS%
4060  FOR I = 1 TO NS%
4070 X1 = X1 + DX:Y1 = Y1 + DY
4080  GOSUB 5000
4090  NEXT I
4100  RETURN
5000  REM  PLOT A POINT AT X1,Y1
5010 XX = X1 * XFAC:YY = Y1 * YFAC
5020 COL =  INT (XX) + 1
5030 ROW =  INT (YY / 6)
5040 XIT% =  INT (YY - (6 * ROW)) + 1
5042  POKE 250,BIT%(COL,ROW)
5044  POKE 251,MASK%(XIT%)
```

```
5046  CALL 24576
5050 BIT%(COL,ROW) =  PEEK (252)
5060  RETURN
```

*Figure A-10 is the graphics program (refer to Figure 7-4) amended for the Apple II computer.*

```
10  PRINT  CHR$ (4);"PR#1"
20  REM ### DEFINE PARAGRAPH SYMBOL IN PLACE OF THE @ SYMBOL
30  PRINT  CHR$ (27); CHR$ (42); CHR$ (1); CHR$ (64); CHR$ (0); CHR$ (6); CHR$
     (25); CHR$ (0); CHR$ (25); CHR$ (118); CHR$ (25); CHR$ (0); CHR$ (127);
     CHR$ (0);
40  REM ### SELECT DOWNLOADABLE CHARACTERS
50  PRINT  CHR$ (27); CHR$ (36); CHR$ (1);
60  REM ### PRINT OUR NEW SYMBOL
70  PRINT "@@@@@@"
```

*Figure A-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the Apple II computer.*

```
5  PRINT  CHR$ (4);"PR#1"
10  REM  ### DEFINE COPYRIGHT SYMBOL IN PLACE OF A @ SYMBOL
20  PRINT  CHR$ (27); CHR$ (42); CHR$ (1); CHR$ (64); CHR$ (0); CHR$ (28);
     CHR$ (34); CHR$ (73); CHR$ (20); CHR$ (65); CHR$ (20); CHR$ (65); CHR$
    (34); CHR$ (20);
30  REM ### SELECT DOWNLOADABLE CHARACTERS
40  PRINT  CHR$ (27); CHR$ (36); CHR$ (1);
50  REM ### PRINT OUR NEW SYMBOL
60  PRINT "COPYRIGHT @ 1983"
```

*Figure A-12 is the downloading copyright program (refer to Figure 8-4) amended for the Apple II computer.*

```
5  PRINT  CHR$ (4);"PR#1"
10  REM ### COPY CHARACTERS INTO RAM
20  PRINT  CHR$ (27); CHR$ (42); CHR$ (0);
30  REM  ### DEFINE COPYRIGHT SYMBOL IN PLACE OF @
40  PRINT  CHR$ (27); CHR$ (42); CHR$ (1); CHR$ (64); CHR$ (0); CHR$ (28);
      CHR$ (34); CHR$ (73); CHR$ (20); CHR$ (65); CHR$ (20); CHR$ (65); CHR$
     (34); CHR$ (20);
50  REM ### SELECT DOWNLOADABLE CHARACTERS
60  PRINT  CHR$ (27); CHR$ (36); CHR$ (1);
70  REM ### PRINT OUR NEW SYMBOL
80  PRINT "COPYRIGHT @ 1983"
```

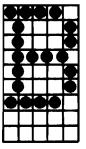
*Figure A-13 is the downloading program presented in Figure 8-5 amended for the Apple II computer.*

```
5  PRINT  CHR$ (4);"PR#1"
10  REM ### COPY CHARACTERS INTO RAM
20  PRINT  CHR$ (27); CHR$ (42); CHR$ (0);
30  REM  ### DEFINE COPYRIGHT SYMBOL IN PLACE OF THE @ SYMBOL
40  PRINT  CHR$ (27); CHR$ (42); CHR$ (1); CHR$ (64); CHR$ (0); CHR$ (28);
      CHR$ (34); CHR$ (73); CHR$ (20); CHR$ (65); CHR$ (20); CHR$ (65); CHR$
     (34); CHR$ (20);
45  REM  ### DEFINE COPYRIGHT SYMBOL IN PLACE OF THE @ SYMBOL
50  PRINT  CHR$ (27); CHR$ (42); CHR$ (1); CHR$ (38); CHR$ (1); CHR$ (28);
      CHR$ (34); CHR$ (73); CHR$ (20); CHR$ (65); CHR$ (20); CHR$ (65); CHR$
     (34); CHR$ (20);
60  REM ### SELECT DOWNLOADABLE CHARACTERS
65  PRINT  CHR$ (27); CHR$ (36); CHR$ (1);
70  REM ### PRINT OUR NEW SYMBOLS
80  PRINT "@&@&@&@&@&@&@&@&@&@&"
```

*Figure A-14 is the downloading program presented in Figure 8-6 amended for the Apple II computer.*

```
5  PRINT  CHR$ (4);"PR#1"
10  REM ### DEFINE MACRO TO BE EMPHASIZED AND EXPANDED TEXT
20  PRINT  CHR$ (27); CHR$ (43); CHR$ (27);"E"; CHR$ (27);"W"; CHR$ (1); CHR$
    (30);
30  REM ### ISSUE MACRO COMMAND AND TRY PRINTING
40  PRINT  CHR$ (27); CHR$ (33);
50  PRINT "EMPHASIZED AND EXPANDED BY MACRO"
```

*Figure A-15 is the macro-commands program (refer to Figure 8-7) amended for the Apple II computer.*



# ATARI 800/400

## The Interface

The best way to hook up your Atari to your Gemini is with the **Universal/Atari Parallel Interface** by Star Micronics. It comes complete with its own cable. You may also use an Atari 850 interface; pin-outs to build a cable for the 850 are listed in this Appendix.

## Setting the DIP Switches

Turn DIP switch number 2-3 and 2-4 "off" on your Gemini, and position switch number 2-1 as you wish.

DIP switch number 2-2 shold be set "off" if you want Gemini to print every time it receives a CR code; set the switch to "on" if you wish to have the carriage return ignored, allowing for printing only when the buffer is full.

## The Use of BASIC

Atari BASIC is somewhat different from the Microsoft BASIC used in this manual. If you do not have the Microsoft BASIC cartridge for your Atari, the following discussion will help you understand the modification necessary for the programs given in this manual. Then you may simply type in the amended programs listed at the back of this appendix.

Send a program listing to the Gemini using a LIST "P" command instead of LLIST (as shown in the sample programs in this manual).

All character-string variables must be dimensional if you are using Atari BASIC. The maximum length of a string is 99 characters.

If you use Atari BASIC, ending the LPRINT command with a semicolon causes Atari to insert spaces in the line until it reaches

40 characters in length. To avoid this problem, you must remember another form of the print command. Instead of

```
LPRINT   CHR$(27);   "5";
```

you need to open a device port, using the OPEN statement:

```
10   OPEN    #7,8,0,"P"
```

Note: #7 tells the Atari to open file number 7, the #8 tells it that you are using file number 7 as an output device; 0 is not used, and the "P" assigns the device to the printer.

Once the file is assigned, you would use the following statement:

```
PRINT   #7;   CHR$(27);"5";
```

At the end of your program, you need to purge the printer buffer and close the file. Use the following statements:

```
PRINT   #7
CLOSE   #7
```

Because Atari uses codes 0-31 for its graphics characters, you may want to add 128 to the ASCII control codes. For example, send CHR$(141) instead of CHR$(13) for a carriage return. As another example, send CHR$(155) instead of CHR$(27) for the start of an escape-function code sequence.

The Atari BASIC does not support the BASIC tab function. If you have further questions, the first step to finding the answer is to consult your Atari user's manual.

## Pin Configurations

The following chart shows the pin-outs for Gemini-10X and Gemini-15X hooked up to the Atari 400/800, using the 850 module parallel interface.

**Parallel**

| Gemini-10X/15X | | Atari (850) | |
|---|---|---|---|
| 1. | $\overline{\text{STROBE}}$ | 1. | $\overline{\text{STROBE}}$ |
| 2. | D1 | 2. | D1 |
| 3. | D2 | 3. | D2 |
| 4. | D3 | 4. | D3 |
| 5. | D4 | 5. | D4 |
| 6. | D5 | 6. | D5 |
| 7. | D6 | 7. | D6 |
| 8. | D7 | 8. | D7 |
| 16. | SG | 11. | SG |
| 32. | $\overline{\text{ERROR}}$ | 12. | $\overline{\text{FAULT}}$ |
| 11. | BUSY | 13. | BUSY |
| 9. | D8 | 15. | D8 |

Following are the program listings amended for the Atari computer. Figure B-2 is a new word processing program similar to the one presented in Chapter 5, but without the use of the single character input function, which the Atari computer does not have. The basic idea of the program is the same; you can input 22 strings of text and each string can be up to 80 characters long.

After typing your text, you can get into command level by entering the word "COM" and hitting Return. In command level, you will have a menu at the top of your screen. From this point, everything works like the word processing program in Chapter 5. You have the option of "ESC" "CHR$" to send codes to the printer; and after that, you can print your text or return to the first level to enter more text.

## Program Listings

```
5 DIM M1$(5),M2$(5),M3$(5)
10 OPEN #4,8,0,"P"
20 PRINT #4;CHR$(27);"@";CHR$(27);"R";CHR$(15);
30 FOR I=1 TO 2
40 IF I=1 THEN PRINT #4;CHR$(14);"   GEMINI STANDARD TYPE STYLES";CHR$(18)
50 IF I=2 THEN PRINT #4;CHR$(14);CHR$(27);"4";"    GEMINI ITALIC TYPE STYLE";CHR$(18)
60 FOR N=1 TO 40:PRINT #4;CHR$(231);CHR$(232);:NEXT N:PRINT #4:PRINT #4
70 PRINT #4;"        17 CPI 12 CPI  10 CPI  8.5 CPI   6 CPI         5 CPI"
80 FOR N=1 TO 80:PRINT #4;CHR$(241);:NEXT N:PRINT #4
90 M1$=" ABC"
92 M2$="X":M2$(LEN(M2$)+1)=CHR$(120)
94 M3$="Y":M3$(LEN(M3$)+1)=CHR$(121)
100 FOR J=1 TO 4
110 IF I=2 THEN PRINT #4;CHR$(27);"4";
120 IF J=1 THEN PRINT #4;CHR$(27);"B";CHR$(3);"NORMAL         ";
130 IF J=2 THEN PRINT #4;CHR$(27);"B";CHR$(3);"DOUBLE STRIKE  ";
140 IF J=3 THEN PRINT #4;CHR$(27);"B";CHR$(3);"EMPHASIZED     ";
150 IF J=4 THEN PRINT #4;CHR$(27);"B";CHR$(3);"DOUBLE STRIKE  ";
160 FOR K=1 TO 6
170 IF K=1 THEN PRINT #4;CHR$(27);"B";CHR$(3);
180 IF K=2 THEN PRINT #4;CHR$(27);"B";CHR$(2);
190 IF K=3 THEN PRINT #4;CHR$(27);"B";CHR$(1);
200 IF K=4 THEN PRINT #4;CHR$(27);"B";CHR$(3);CHR$(14);
210 IF K=5 THEN PRINT #4;CHR$(27);"B";CHR$(2);CHR$(14);
220 IF K=6 THEN PRINT #4;CHR$(27);"B";CHR$(1);CHR$(14);
230 IF J=2 THEN PRINT #4;CHR$(27);"G";
240 IF J=3 THEN PRINT #4;CHR$(27);"E";
250 IF J=4 THEN PRINT #4;CHR$(27);"G";CHR$(27);"E";
260 PRINT #4;M1$;CHR$(27);"S";CHR$(0);M2$;CHR$(27);"S";CHR$(1);M3$;
265 PRINT #4;CHR$(27);"T";CHR$(18);
300 IF K>3 THEN PRINT #4;CHR$(18);
310 NEXT K
320 IF J=4 THEN GOSUB 500
340 PRINT #4;CHR$(27);"@"
350 IF J=2 THEN PRINT #4;CHR$(10);
360 NEXT J
370 FOR M=1 TO 5:PRINT #4:NEXT M
380 NEXT I
400 CLOSE #4
```

```
410 END
500 PRINT #4;CHR$(27);"A";CHR$(9);CHR$(10);
510 PRINT #4;CHR$(27);"B";CHR$(3);
520 PRINT #4;"     + ";CHR$(13);"EMPHASIZED ";
530 RETURN
```

*Figure B-1 is the font styles program (refer to Figure 5-1) amended for the Atari computer.*

```
5 DIM A$(80),B(22,80)
6 DIM C(22),D$(10),E$(10),F$(10),S(10)
7 N=1
10 GRAPHICS 0
15 GOSUB 2000
20 INPUT A$
25 IF A$="STOP" THEN 160
26 IF A$="COM" THEN 500
30 GOSUB 1000
40 GOTO 20
100 IF N=1 THEN 160
105 FOR I=1 TO N-1
110 FOR J=1 TO C(I)
120 PRINT CHR$(B(I,J));
130 NEXT J
140 PRINT
150 NEXT I
160 END
500 REM
505 GRAPHICS 0:PRINT "ENTER YOUR COMMAND....."
507 PRINT "P)RINT Q)UIT  E)SC C)HR$"
509 PRINT "_____"
510 INPUT D$
520 IF D$="E" THEN GOSUB 700
530 IF D$="C" THEN GOSUB 600
540 IF D$="Q" THEN GOTO 7
545 IF D$="P" THEN GOSUB 800
550 GOTO 500
600 REM
605 PRINT "ENTER YOUR CODE CHR$(  )";
610 INPUT L
```

```
620 OPEN #4,8,0,"P"
630 PRINT #4,CHR$(L)
640 CLOSE #4
650 RETURN
700 REM
710 PRINT "ENTER YOUR ESC CODE";
720 INPUT E$
725 J=1
730 FOR I=1 TO LEN(E$)
732 X=ASC(E$(I)):IF X=32 THEN 740
735 S(J)=X:J=J+1
740 NEXT I
750 OPEN #4,8,0,"P"
751 IF J=2 THEN 760
752 AS=2
755 K=VAL(E$(AS,J-1))
760 PRINT #4;CHR$(27);CHR$(S(1));
770 PRINT #4;CHR$(K)
780 CLOSE #4
790 RETURN
800 REM
803 IF N=1 THEN 870
805 OPEN #4,8,0,"P"
810 FOR I=1 TO N-1
820 FOR J=1 TO C(I)
830 PRINT #4;CHR$(B(I,J));
840 NEXT J
850 PRINT #4
860 NEXT I
865 CLOSE #4
870 RETURN
1000 REM
1010 FOR J=1 TO LEN(A$)
1020 B(N,J)=ASC(A$(J))
1030 NEXT J
1040 C(N)=LEN(A$)
1045 N=N+1
1050 RETURN
2000 REM
2010 PRINT "YOU CAN TYPE UP TO 22 STRINGS AND "
2020 PRINT "EACH STRING CAN BE 80 CHARACTERS"
2030 PRINT "LONG.  TYPE 'COM' TO GOTO COMMAND"
```

```
2040 PRINT "LEVEL...AND 'STOP' TO GET OUT"
2050 PRINT "-----------------------------------"
2060 RETURN
```

*Figure B-2 is the word processing program (refer to Figure 5-4) amended for the Atari computer.*

```
5 DIM A$(30),B$(12)
10 OPEN #4,8,0,"P"
20 A$="                                "
25 B$="            "
30 PRINT #4;A$;"HIGH RESOLUTION VERTICAL SPACING"
35 PRINT #4;A$;"USING THE ESC 3 N  CONTROL CODE"
40 FOR I=1 TO 30
50 IF I=13 THEN 140
60 M=INT(144/(6*I)):IF M<2 THEN M=2
70 PRINT #4;CHR$(27);CHR$(51);CHR$(I);
80 FOR J=0 TO M
90 IF J=M-1 THEN PRINT #4;I;"/144 INCH SPACING";B$;:GOTO 110
100 PRINT #4;A$;
110 FOR N=1 TO 40:PRINT #4;CHR$(241);:NEXT N:PRINT #4
120 NEXT J
140 NEXT I
150 PRINT #4;CHR$(27);"@"
160 CLOSE #4
170 END
```

*Figure B-3 is the vertical spacing program (refer to Figure 6-5) amended for the Atari computer.*

```
10 OPEN #4,8,0,"P"
20 PRINT #4;CHR$(27);"@";CHR$(27);"A";CHR$(6)
30 FOR LINE=1 TO 2
40 PRINT #4;CHR$(27);"K";CHR$(114);CHR$(0);
50 FOR SPACE=1 TO 100
55 PRINT #4;CHR$(0);:NEXT SPACE
60 FOR COLM=1 TO 14
70 READ C:PRINT #4;CHR$(C);:NEXT COLM
80 PRINT #4:NEXT LINE
```

```
90 DATA 14,16,34,64,140,140,128,128,140,140,64,34,16,14
100 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
110 PRINT #4;CHR$(27);"@"
120 FOR I=1 TO 5:PRINT #4:NEXT I
130 CLOSE #4
140 END
```

*Figure B-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the Atari computer.*

```
10 OPEN #4,8,0,"P"
20 PRINT #4;CHR$(163);CHR$(161);
30 PRINT #4;CHR$(27);"A";CHR$(6)
40 PRINT #4;CHR$(162);CHR$(160);
50 PRINT #4;CHR$(27);"2"
60 CLOSE #4:END
```

*Figure B-5 is the circle program presented in Chapter 7, amended for the Atari computer.*

```
10 OPEN #4,8,0,"P"
20 PRINT #4;" ";CHR$(163);
30 PRINT #4;CHR$(27);"A";CHR$(6)
40 PRINT #4;CHR$(160)
50 CLOSE #4:END
```

*Figure B-6 is the tilde program presented in Chapter 7 amended for the Atari computer.*

```
10 OPEN #4,8,0,"P"
20 PRINT #4,"  N";CHR$(13);
30 PRINT #4,"  ";CHR$(164);CHR$(13);
40 PRINT #4,"W";CHR$(166);CHR$(32);CHR$(167);"E";CHR$(13);
50 PRINT #4,"  ";CHR$(165);CHR$(13);
60 PRINT #4,"  S"
70 CLOSE #4
80 END
```

*Figure B-7 is the map program presented in Chapter 7 amended for the Atari computer.*

```
10 OPEN #4,8,0,"P"
15 PRINT #4;CHR$(27);"U";CHR$(1)
20 PRINT #4;CHR$(27);"A";CHR$(6)
30 FOR J=1 TO 9
40 FOR I=1 TO 12
50 READ A
60 PRINT #4;CHR$(A);
70 NEXT I
80 PRINT #4
90 NEXT J
100 PRINT #4;CHR$(27);"@"
110 CLOSE #4
120 END
130 DATA 239,224,224,224,239,224,224,224,224,224,224,224
140 DATA 239,224,224,224,239,224,224,224,224,224,224,224
150 DATA 239,224,224,224,239,224,239,224,224,224,239,224
160 DATA 239,224,239,224,239,224,234,233,224,234,233,224
170 DATA 239,234,239,233,239,224,224,239,224,239,224,224
180 DATA 239,239,224,239,239,224,224,234,239,233,224,224
190 DATA 234,233,224,234,233,224,224,224,239,224,224,224
200 DATA 224,224,224,224,224,224,224,234,233,224,224,224
210 DATA 224,224,224,224,224,224,224,239,224,224,224,224
```

*Figure B-8 is the "Wy" program (refer to Figure 7-1) amended for the Atari computer.*

```
10 OPEN #4,8,0,"P"
20 PRINT #4;CHR$(27);"A";CHR$(6);CHR$(15)
25 PRINT #4;CHR$(27);"U";CHR$(1)
30 FOR K=1 TO 3
40 PRINT #4;CHR$(27);"B";CHR$(K)
50 RESTORE
60 FOR J=1 TO 9
70 FOR I=1 TO 12
80 READ A:PRINT #4;CHR$(A);
90 NEXT I
95 PRINT #4
100 NEXT J
110 PRINT #4
120 NEXT K
130 PRINT #4;CHR$(27);"@"
```

```
140 CLOSE #4
150 END
200 DATA 224,224,224,224,224,224,224,224,224,245,224,224
210 DATA 224,224,224,224,224,224,224,224,252,245,224,224
220 DATA 224,224,224,245,224,224,224,252,239,245,224,224
230 DATA 224,224,252,245,224,224,252,239,239,245,224,224
240 DATA 224,252,239,245,224,252,239,239,239,245,224,224
250 DATA 252,239,239,245,252,239,239,239,239,245,224,224
260 DATA 241,241,241,249,241,241,241,241,241,249,224,224
270 DATA 239,239,239,239,239,239,239,239,239,239,239,251
280 DATA 253,239,239,239,239,239,239,239,239,239,251,224
```

*Figure B-9 is the boat program (refer to Figure 7-2) amended for the Atari computer.*

```
2 GRAPHICS 0
3 PRINT " ":PRINT " ":PRINT " "
4 PRINT " THIS PROGRAM TAKES ABOUT"
5 PRINT " 4 MINUTES TO RUN, PLEASE"
6 PRINT " TURN ON YOUR PRINTER AND "
7 PRINT " STAND BY ....."
8 PRINT " ":PRINT " ":PRINT " "
10 FOR I=1536 TO 1553
20 READ D
30 POKE I,D
40 NEXT I
50 DATA 104,104,133,205,104,133,204
60 DATA 104,5,205,133,213,104,5,204
70 DATA 133,212,96
100 REM MICRO-PLOT
110 DIM BIT(76,12),MASK(6)
120 DIM A$(76)
130 DIM B$(10)
132 FOR I=0 TO 11
133 FOR J=1 TO 76
134 BIT(J,I)=0
135 NEXT J
136 NEXT I
1000 REM SET PROGRAM CONSTANT
1010 MASK(1)=64:MASK(4)=8
1020 MASK(2)=32:MASK(5)=4
```

```
1030 MASK(3)=16:MASK(6)=2
1040 LX=20:LY=20
1050 XFAC=72/LX:YFAC=75/LY
2000 REM PLOT CURVE
2010 LET RAD=9
2020 X1=19:Y1=10
2030 FOR AN6=0 TO 360 STEP 10
2040 R1=AN6*6.28/360
2050 X2=RAD*COS(R1)+10
2055 Y2=RAD*SIN(R1)+10
2060 GOSUB 4000
2070 NEXT AN6
2080 FOR AN6=0 TO 360 STEP 15
2090 R1=AN6*6.28/360
2100 R2=(AN6+150)*6.28/360
2110 X1=RAD*COS(R1)+10
2115 Y1=RAD*SIN(R1)+10
2120 X2=RAD*COS(R2)+10
2125 Y2=RAD*SIN(R2)+10
2130 GOSUB 4000
2140 NEXT AN6
3000 REM SEND BIT IMAGE TO PRINTER
3005 OPEN #4,8,0,"P"
3010 PRINT #4;CHR$(27);"A";CHR$(6)
3020 FOR ROW=0 TO 11
3040 PRINT #4;CHR$(27);"K";CHR$(119);CHR$(1);
3050 FOR COL=1 TO 75
3052 FOR LL=1 TO 5
3054 RE=INT(BIT(COL,ROW))
3060 PRINT #4;CHR$(RE);
3065 NEXT LL
3070 NEXT COL
3080 PRINT #4
3090 NEXT ROW
3100 PRINT #4;CHR$(27);"A";CHR$(12)
3105 PRINT #4," ":PRINT #4," "
3110 CLOSE #4
3150 END
4000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
4010 XL=X2-X1:YL=Y2-Y1
4020 NX=ABS(XL*XFAC):NY=ABS(YL*YFAC)
```

```
4030 IF NX<NY THEN NX=NY
4040 NS=INT(NX+1)
4050 DX=XL/NS
4055 DY=YL/NS
4060 FOR I=1 TO NS
4070 X1=X1+DX:Y1=Y1+DY
4080 GOSUB 5000
4090 NEXT I
4100 RETURN
5000 REM PLOT A POINT AT X1,Y1
5010 XX=X1*XFAC:YY=Y1*YFAC
5020 COL=INT(XX)+1
5030 ROW=INT(YY/6)
5040 XIT=INT(YY-ROW*6)+1
5050 A1=BIT(COL,ROW)
5060 A2=MASK(XIT)
5070 BIT(COL,ROW)=USR(1536,A1,A2)
5080 RETURN
```

*Figure B-10 is the graphics program (refer to Figure 7-4) amended for the Atari computer.*

```
10 DIM E$(5)
20 REM DEFINE ESCAPE CHARACTERS
25 E$=CHR$(27)
30 OPEN #4,8,0,"P"
35 REM DEFINE PARAGRAPH SYMBOL IN PLACE @
40 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(6);
45 PRINT #4;CHR$(9);CHR$(0);CHR$(9);CHR$(118);CHR$(9);CHR$(0);
47 PRINT #4;CHR$(127);CHR$(0)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT #4;E$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT #4;"@@@@@@"
90 CLOSE #4:END
```

*Figure B-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the Atari computer.*

```
10 DIM E$(5)
20 REM DEFINE ESCAPE CHARACTERS
25 E$=CHR$(27)
30 OPEN #4,8,0,"P"
35 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT #4;CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT #4;CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT #4;E$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT #4;"COPY RIGHT @ 1983"
90 CLOSE #4:END
```

*Figure B-12 is the downloading copyright program (refer to Figure 8-4) amended for the Atari computer.*

```
10 DIM E$(5)
15 OPEN #4,8,0,"P"
20 REM DEFINE ESCAPE CHARACTERS
25 E$=CHR$(27)
27 REM COPY CHARACTERS TO RAM
28 PRINT #4;E$;CHR$(42);CHR$(0)
35 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT #4;CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT #4;CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT #4;E$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT #4;"COPY RIGHT @ 1983"
90 CLOSE #4:END
```

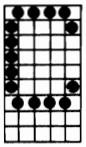
*Figure B-13 is the downloading program presented in Figure 8-5 amended for the Atari computer.*

```
10 DIM E$(5)
15 OPEN #4,8,0,"P"
20 REM DEFINE ESCAPE CHARACTERS
25 E$=CHR$(27)
27 REM COPY CHARACTERS TO RAM
28 PRINT #4;E$;CHR$(42);CHR$(0)
35 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT #4;CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT #4;CHR$(65);CHR$(34);CHR$(20)
49 REM DEFINE COPYRIGHT SYMBOL IN PLACE &
50 PRINT #4;E$;CHR$(42);CHR$(1);CHR$(38);CHR$(1);CHR$(28);
55 PRINT #4;CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
57 PRINT #4;CHR$(65);CHR$(34);CHR$(20)
59 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT #4;E$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT #4;"@&@&@&@&@&@&@&@"
90 CLOSE #4:END
```

*Figure B-14 is the downloading program presented in Figure 8-6 amended for the Atari computer.*

```
5 DIM E$(5)
10 REM DEFINE ESC CHARACTER
20 E$=CHR$(27)
30 OPEN #4,8,0,"P"
40 PRINT #4;E$;CHR$(43);E$;"E";E$;"W";CHR$(1);CHR$(30)
50 REM ISSUE MACRO COMMAND
60 PRINT #4;E$;CHR$(33)
70 PRINT #4;"EMPHASIZED AND EXPANDED BY MACRO"
80 CLOSE #4:END
```

*Figure B-15 is the macro-commands program (refer to Figure 8-7) amended for the Atari computer.*



# COMMODORE VIC-20 & C-64

## The Connecting Cable

Star Micronics, Inc. (the makers of Gemini), has created a special Centronics parallel interface especially for your VIC-20 and C-64 computers. It is called the **Universal/Commodore Parallel Interface.**

If you haven't already done so, contact your Gemini dealer about obtaining and installing the interface device and cable.

## Setting the DIP Switches

The only DIP switch that must be set in one position or the other is switch number 2-3, which must be "off" on your Gemini. Switch number 2-2 should be set to "off" if you want Gemini to print every time it receives a CR code; set the switch "on" if you wish to have the carriage return ignored and printing only when the buffer is full.

## The Use of BASIC

We recommend the use of Commodore's Advanced BASIC Program. If you encounter difficulty in running the programs presented in this User's Manual, consult the documentation that supports the Advanced BASIC Program for Commodore.

## Program Listings

```
20 E$=CHR$(27):LF$=CHR$(10):FF$=CHR$(12):CR$=CHR$(13)
30 PC$=E$+"B"+CHR$(1):EL$=E$+"B"+CHR$(2):CM$=E$+"B"+CHR$(3)
40 SP$=E$+"S"+CHR$(0):SB$=E$+"S"+CHR$(1):CS$=E$+"T"
50 IT$=E$+CHR$(52)
60 WD$=CHR$(14):CW$=CHR$(20) :PS$=E$+"@"
70 EM$=E$+"E":CE$=E$+"F":DB$=E$+"G":CD$=E$+"H"
80 OPEN4,4,5:PRINT#4,PS$;E$;"R";CHR$(15);
90 FOR I=1 TO 2
100 IF I=1 THEN PRINT#4,WD$;"    GEMINI STANDARD TYPE STYLES";CW$;LF$;
110 IF I=2 THEN PRINT#4,IT$;WD$;"     GEMINI ITALIC TYPE STYLE";
115 PRINT#4,CW$;LF$
120 FOR N=1 TO 40:PRINT#4,CHR$(231);CHR$(232);:NEXT N:PRINT#4:PRINT#4
130 PRINT#4,"         17 CPI 12 CPI  10 CPI  8.5 CPI  6 CPI     5 CPI"
140 FOR N=1 TO 80:PRINT#4,CHR$(241);:NEXT N:PRINT#4
150 MS6$(1)=" ABC":MS6$(2)="X"+CHR$(120):MS6$(3)="Y"+CHR$(121)
160 FOR J=1 TO 4
170 IF I=2 THEN PRINT#4,IT$;
180 IF J=1 THEN PRINT#4,CM$;"NORMAL         ";
190 IF J=2 THEN PRINT#4,CM$ ;"DOUBLE STRIKE  ";
200 IF J=3 THEN PRINT#4,CM$ ;"EMPHASIZED     ";
210 IF J=4 THEN PRINT#4,CM$ ;"DOUBLE STRIKE  ";
220 FOR K=1 TO 6
230 IF K=1 THEN PRINT#4,CM$;
240 IF K=2 THEN PRINT#4,EL$;
250 IF K=3 THEN PRINT#4,PC$;
260 IF K=4 THEN PRINT#4,CM$;WD$;
270 IF K=5 THEN PRINT#4,EL$;WD$;
280 IF K=6 THEN PRINT#4,PC$;WD$;
290 IF J=2 THEN PRINT#4,DB$;
300 IF J=3 THEN PRINT#4,EM$;
310 IF J=4 THEN PRINT#4,DB$;EM$;
320 PRINT#4,MS6$(1);SP$;MS6$(2);CS$;SB$;MS6$(3);CS$;CD$;
330 IF K>3 THEN PRINT#4,CW$;
340 NEXT K
350 IF J=4 THEN PRINT#4,E$;"A";CHR$(9);LF$;CM$;"     + ";CR$;"EMPHASIZED ";
360 PRINT#4,PS$
365 IF J=2 THEN PRINT#4,LF$;
```

```
370 NEXT J
380 FOR M=1 TO 5:PRINT#4,LF$:NEXT M
390 NEXT I
400 END
```

*Figure C-1 is the font styles program (refer to Figure 5-1) amended for the Commodore computer.*

```
10 DIM A$(22,40'
12 C$=CHR$(81)
20 PRINT CHR$(147):X=0:Y=0:Z=1024
30 GOSUB 150:POKE Z+X*40+Y,ASC(C$)
40 GET B$:IF B$=""THEN 40
50 IF B$=CHR$(5)THEN PRINT CHR$(147):END
60 IF B$=CHR$(18) THEN GOSUB 200:GOTO 20
70 IF B$=CHR$(16) THEN GOTO 220
80 IF B$=CHR$(13) THEN GOSUB 700:GOTO 40
90 IF B$=CHR$(4)THEN GOSUB 130:GOTO 40
100 B=ASC(B$):IF B>64 THEN B=B-64
102 POKE Z+X*40+Y,B:A$(X,Y)=B$:Y=Y+1
104 IF Y=40 THEN A$(X,Y)=CHR$(13):X=X+1:Y=0:GOTO 40
120 POKE Z+X*40+Y,ASC(C$):GOTO 40
130 POKE Z+X*40+Y,ASC(" "):Y=Y-1:IF Y<0 THEN Y=0
135 A$(X,Y)=""
140 POKE Z+X*40+Y,ASC(C$):RETURN
150 M$="CTRL P)RINT  D)ELETE  R)ESTART  E)XIT"
160 F=1984:GOSUB 800
170 RETURN
200 PRINT CHR$(147):M$="PLEASE STAND BY"
205 F=1555:GOSUB 800
210 FOR I=0 TO X:FOR J=1 TO 40:A$(I,J)="":NEXT J:NEXT I
215 RETURN
220 OPEN4,4,5:PRINT#4,CHR$(27);"@":CLOSE4
230 PRINT CHR$(147):M$="C OMMAND  P)RINT  R)ETURN":F=1984:GOSUB 800
```

```
250 GET G$:IF G$=""THEN 250
260 IF G$="P" THEN GOSUB 310
270 IF G$="C" THEN GOSUB 320:GOTO 230
280 IF G$="R" THEN GOSUB 300:GOTO 30
290 GOTO 250
300 PRINT CHR$(147):GOSUB 900:RETURN
310 OPEN4,4,5:CMD4
312 FOR I=0 TO X:FOR J=0 TO 40
314 PRINT A$(I,J);:NEXT J
316 NEXT I:PRINT#4:CLOSE4:RETURN
320 M$="                                ":F=1984:GOSUB 800
330 M$="E) ESC  C) CHR$  Q) END COMMAND":F=1984:GOSUB 800
340 ST$=""
360 GET G$:IF G$=""THEN 360
370 IF G$="E" THEN GOSUB 420:GOTO 400
380 IF G$="C" THEN GOSUB 530:GOTO 400
390 IF G$="Q" THEN GOTO 410
400 GOTO 360
410 RETURN
420 L=0:M$=" ESC ":F=1910:GOSUB 800
425 F=1915
430 GET G$:IF G$=""THEN 430
440 IF G$=CHR$(13) THEN 480
450 M$=G$+" ":F=F+2:GOSUB 800
470 S$(L)=G$:L=L+1:GOTO 430
480 IF L>2 THEN GOSUB 600:
490 IF L=2 THEN GOSUB 650
500 IF L=1 THEN ST$=CHR$(27)+S$(0)
510 OPEN4,4,5:PRINT#4,ST$:CLOSE4:ST$=""
515 FOR J=1 TO L:ST$(J)="":NEXT J:L=0
520 M$="                        ":F=1910:GOSUB 800:RETURN
530 M$=" CHR$ ":F=1910:GOSUB 800:ST$=""
535 F=1916
540 GET G$:IF G$=""THEN 540
550 IF G$=CHR$(13) THEN 570
560 M$=G$:F=F+1:GOSUB 800:ST$=ST$+G$:GOTO 540
570 M$="                              ":F=1910:GOSUB 800:
580 OPEN4,4,5:PRINT#4,CHR$(VAL(ST$));:CLOSE4:ST$="":RETURN
600 REM
610 FOR J=1 TO L:D$=D$+S$(J):NEXT J
620 ST$=CHR$(27)+S$(0)+CHR$(VAL(D$))
```

```
630 D$=""
640 RETURN
650 ST$=CHR$(27)+S$(0)+CHR$(VAL(S$(1)))
660 RETURN
700 REM
710 A$(X,Y)=B$:
720 POKE Z+X*40+Y,ASC(" ")
730 X=X+1:Y=0
740 POKE Z+X*40+Y,ASC(C$)
750 RETURN
800 FOR I=F TO F-1+LEN(M$)
810 J=I-(F-1)
820 M=ASC(MID$(M$,J,1))
830 IF M>64 THEN M=M-64
840 POKE I,M
850 NEXT I
860 RETURN
900 REM
910 FOR I=0 TO X:FOR J=0 TO 39
915 IF A$(I,J)=""THEN 950
920 A=ASC(A$(I,J))
925 IF A=13 THEN W=W+40:GOTO 950
930 IF A>64 THEN A=A-64
935 W=Z+I*40+J
940 POKE W,A
950 NEXT J
960 NEXT I
970 RETURN
```

*Figure C-2 is the word processing program (refer to Figure 5-4) amended for the Commodore computer. NOTE: The only way to move the cursor around on the Commodore computer is to use the screen memory map. We can specify the location on the screen and then "POKE" the character into memory. Because of this limitation, we had to use "POKE" instead of "PRINT" statement for the Commodore computer. However, when we "POKE" characters into screen memory, the color will be light so you must make sure that the background color on your screen is dark.*

```
10 OPEN4,4,5:CMD4
20 PRINT TAB(30);"HIGH RESOLUTION VERTICAL SPACING"
30 PRINT TAB(30);"USING THE ESC 3 N  CONTROL CODE"
40 FOR I=1 TO 30
50 IF I=13 THEN GOTO 140
60 M=INT(144/(6*I)):IF M<2 THEN M=2
70 PRINT CHR$(27);CHR$(51);CHR$(I);
80 FOR J=0 TO M
90 IF J=M-1 THEN PRINT I;"/144 INCH SPACING";TAB(10);:GOTO 110
100 PRINT TAB(30);
110 FOR N=1 TO 40:PRINT CHR$(241);:NEXT N:PRINT
120 NEXT J
140 NEXT I
150 PRINT CHR$(27);"@"
160 PRINT#4:CLOSE4
170 END
```

*Figure C-3 is the vertical spacing program (refer to Figure 6-5) amended for the Commodore computer.*

```
10 OPEN4,4,5:CMD4
20 PRINT CHR$(27);"@";CHR$(27);"A";CHR$(6
30 FOR LINE=1 TO 2
40 PRINT CHR$(27);"K";CHR$(114);CHR$(0);
50 FOR SPACE=1 TO 100
55 PRINT CHR$(0);:NEXT SPACE
60 FOR COLM=1 TO 14
70 READ C:PRINT CHR$(C);:NEXT COLM
80 PRINT :NEXT LINE
90 DATA 14,16,34,64,140,140,128,128,140,140,64,34,16,14
100 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
110 PRINT CHR$(27);"@":PRINT:PRINT:PRINT
120 PRINT#4:CLOSE4
130 END
```

*Figure C-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the Commodore computer.*

```
10 OPEN4,4,5:CMD4
20 PRINT CHR$(163);CHR$(161);
30 PRINT CHR$(27);"A";CHR$(6)
40 PRINT CHR$(162);CHR$(160);
50 PRINT CHR$(27);"2"
60 PRINT#4:CLOSE4:END
```

*Figure C-5 is the circle program presented in Chapter 7, amended for the Commodore computer.*

```
10 OPEN4,4,5:CMD4
20 PRINT TAB(1);CHR$(163);CHR$(27);"A";CHR$(6)
30 PRINT CHR$(160)
40 PRINT#4:CLOSE4
50 END
```

*Figure C-6 is the tilde program presented in Chapter 7 amended for the Commodore computer.*

```
10 OPEN4,4,5:CMD4
20 PRINT TAB(20);"N";CHR$(13);
30 PRINT TAB(20);CHR$(164);CHR$(13);
40 PRINT TAB(18);"W";CHR$(166);CHR$(32);CHR$(167);"E";CHR$(13);
50 PRINT TAB(20);CHR$(165);CHR$(13);
60 PRINT TAB(20);"S"
70 PRINT#4:CLOSE4
```

*Figure C-7 is the map program presented in Chapter 7 amended for the Commodore computer.*

```
10 OPEN4,4,5:CMD4
15 PRINT CHR$(27);"U";CHR$(1)
20 PRINT CHR$(27);"A";CHR$(6)
30 FOR J=1 TO 9
40 FOR I=1 TO 12
50 READ A
60 PRINT CHR$(A);
70 NEXT I
80 PRINT
90 NEXT J
100 PRINT CHR$(27);"@":
110 PRINT#4:CLOSE4
120 END
130 DATA 239,224,224,224,239,224,224,224,224,224,224,224
140 DATA 239,224,224,224,239,224,224,224,224,224,224,224
150 DATA 239,224,224,224,239,224,239,224,224,224,239,224
160 DATA 239,224,239,224,239,224,234,233,224,234,233,224
170 DATA 239,234,239,233,239,224,224,239,224,239,224,224
180 DATA 239,239,224,239,239,224,224,234,239,233,224,224
190 DATA 234,233,224,234,233,224,224,224,239,224,224,224
200 DATA 224,224,224,224,224,224,224,234,233,224,224,224
210 DATA 224,224,224,224,224,224,224,239,224,224,224,224
```

*Figure C-8 is the "Wy" program (refer to Figure 7-1) amended for the Commodore computer.*

```
10 OPEN4,4,5:CMD4
20 PRINT CHR$(27);"A";CHR$(6);CHR$(15)
25 PRINT CHR$(27);"U";CHR$(1)
30 FOR K=1 TO 3
40 PRINT CHR$(27);"B";CHR$(K)
50 RESTORE
60 FOR J=1 TO 9
70 FOR I=1 TO 12
80 READ A:PRINT CHR$(A);
90 NEXT I
95 PRINT
100 NEXT J
110 PRINT
```

```
120 NEXT K
130 PRINT CHR$(27);"@"
140 PRINT#4:CLOSE4
150 END
200 DATA 224,224,224,224,224,224,224,224,224,245,224,224
210 DATA 224,224,224,224,224,224,224,224,252,245,224,224
220 DATA 224,224,224,245,224,224,224,252,239,245,224,224
230 DATA 224,224,252,245,224,224,252,239,239,245,224,224
240 DATA 224,252,239,245,224,252,239,239,239,245,224,224
250 DATA 252,239,239,245,252,239,239,239,239,245,224,224
260 DATA 241,241,241,249,241,241,241,241,241,249,224,224
270 DATA 239,239,239,239,239,239,239,239,239,239,239,251
280 DATA 253,239,239,239,239,239,239,239,239,239,251,224
```

*Figure C-9 is the boat program (refer to Figure 7-2) amended for the Commodore computer.*

```
50 PRINT CHR$(147)
60 FOR I=1 TO 5:PRINT "/":NEXT I
70 PRINT "THIS  PROGRAM  TAKES ABOUT TWO MINUTES"
80 PRINT "TO RUN, SO PLEASE TURN ON YOUR PRINTER"
90 PRINT "AND STAND BY........................."
100 REM MICRO-PLOT
110 DIM BIT%(76,12)
1000 REM SET PROGRAM CONSTANTS
1010 MASK%(1) = 64   : MASK%(4) = 8
1020 MASK%(2) = 32   : MASK%(5) = 4
1030 MASK%(3) = 16   : MASK%(6) = 2
1040 LX = 20         : LY = 20
1050 XFAC = 72/LX    : YFAC = 75/LY
2000 REM PLOT CURVE
2010 RAD = 9
2020 X1 = 19         : Y1 = 10
2030 FOR ANG = 0 TO 360 STEP 10
2040 R1 = ANG*6.28/360
2050 X2 = RAD*COS(R1)+10 : Y2 = RAD*SIN(R1)+10
2060 GOSUB 4000
2070 NEXT ANG
```

```
2080 FOR ANG = 0 TO 360 STEP 15
2090 R1 = ANG*6.28/360
2100 R2 = (ANG+150)*6.28/360
2110 X1 = RAD*COS(R1)+10 : Y1 = RAD*SIN(R1)+10
2120 X2 = RAD*COS(R2)+10 : Y2 = RAD*SIN(R2)+10
2130 GOSUB 4000
2140 NEXT ANG
3000 REM SEND BIT IMAGE MAP TO PRINTER
3005 OPEN 4,4,5
3010 PRINT#4, CHR$(27) "A" CHR$(6)
3020 FOR ROW = 0 TO 11
3025 PRINT#4,CHR$(27);"K";CHR$(119);CHR$(1);
3050 FOR COL = 1 TO 75
3055 FOR LL=1 TO 5
3060 PRINT#4,CHR$(BIT%(COL,ROW));
3065 NEXT LL
3070 NEXT COL
3080 PRINT#4,""
3090 NEXT ROW
3100 PRINT#4 : CLOSE4
3110 END
4000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
4010 XL = X2 - X1      : YL = Y2 - Y1
4020 NX = ABS(XL*XFAC) : NY = ABS(YL*YFAC)
4030 IF NX < NY THEN NX = NY
4040 NS% = INT(NX+1)
4050 DX = XL/NS%       : DY = YL/NS%
4060 FOR I = 1 TO NS%
4070 X1 = X1 + DX      : Y1 = Y1 + DY
4080 GOSUB 5000
4090 NEXT I
4100 RETURN
5000 REM PLOT A POINT AT X1,Y1
5010 XX = X1 * XFAC    : YY = Y1 * YFAC
5020 COL% = INT(XX)+1
5030 ROW% = INT(YY/6)
5040 XIT% = INT(YY - ROW% * 6) + 1
5050 BIT%(COL%,ROW%) = BIT%(COL%,ROW%) OR MASK%(XIT%)
5060 RETURN
```

*Figure C-10 is the graphics program (refer to Figure 7-4) amended for the Commodore computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE PARAGRAPH SYMBOL IN PLACE @
35 OPEN4,4,5:CMD4
40 PRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(6);
45 PRINT CHR$(9);CHR$(0);CHR$(9);CHR$(118);CHR$(9);CHR$(0);
47 PRINT CHR$(127);CHR$(0)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT "@@@@@@"
90 PRINT#4:CLOSE4:END
```

*Figure C-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the Commodore computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
35 OPEN4,4,5:CMD4
40 PRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT "COPY RIGHT @ 1983"
90 PRINT#4:CLOSE4:END
```

*Figure C-12 is the downloading copyright program (refer to Figure 8-4) amended for the Commodore computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
22 OPEN4,4,5:CMD4
24 REM COPY CHARACTERS TO RAM
25 PRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT CHR$(65);CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT "COPY RIGHT @ 1983"
90 PRINT#4:CLOSE4:END
```

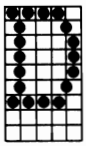
*Figure C-13 is the downloading program presented in Figure 8-5 amended for the Commodore computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
22 OPEN4,4,5:CMD4
24 REM COPY CHARACTERS TO RAM
25 PRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE @
40 PRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 PRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
47 PRINT CHR$(65);CHR$(34);CHR$(20)
49 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF &
50 PRINT ESC$;CHR$(42);CHR$(1);CHR$(38);CHR$(1);CHR$(28);
55 PRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
57 PRINT CHR$(65);CHR$(34);CHR$(20)
59 REM SELECT DOWNLOADABLE CHARACTERS
60 PRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 PRINT "@&@&@&@&@&@&@&@&"
90 PRINT#4:CLOSE4:END
```

*Figure C-14 is the downloading program presented in Figure 8-6 amended for the Commodore computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM CONNECT TO PRINTER
40 OPEN4,4,5:CMD4
50 PRINT ESC$;CHR$(43);ESC$;"E";ESC$;"W";CHR$(1);CHR$(30)
60 REM ISSUE MACRO COMMAND AND TRY PRINTING
65 PRINT ESC$;CHR$(33)
70 PRINT "EMPHASIZED AND EXPANDED BY MACRO"
80 PRINT#4:CLOSE4
90 END
```

*Figure C-15 is the macro-commands program (refer to Figure 8-7) amended for the Commodore computer.*



# IBM PERSONAL COMPUTER

## The Interface Board

Install the IBM parallel interface board according to the directions that accompany the product, which you can obtain from your Gemini dealer or from an IBM salesperson.

Use the Gemini cable provided by your dealer or use cable information at the end of this appendix.

## Setting the DIP Switches

DIP switches 2-3 and 2-4 should be turned "off" on your Gemini, and the other two may be set according to your operational requirements.

Switch 2-2 should be set "off" if you want Gemini to print every time it receives a CR code; set it "on" if you wish to have the carriage return ignored and printing only when the buffer is full.

## The Use of BASIC

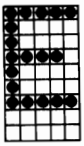
All the example programs provided in this User's Manual are written in BASIC for the IBM Personal Computer. You may run them in IBM Microsoft BASIC or in IBM Microsoft Advanced BASIC (BASICA).

## Pin Configurations

The following chart shows the pin-outs for the Gemini-10X and Gemini-15X, interfaced with the IBM Personal Computer, using a parallel interface board.

**Parallel**

| Gemini-10X/15X | | IBM 25-Pin | |
|---|---|---|---|
| 1. | $\overline{\text{STROBE}}$ | 1. | $\overline{\text{STROBE}}$ |
| 2. | D1 | 2. | D0 |
| 3. | D2 | 3. | D1 |
| 4. | D3 | 4. | D2 |
| 5. | D4 | 5. | D3 |
| 6. | D5 | 6. | D4 |
| 7. | D6 | 7. | D5 |
| 8. | D7 | 8. | D6 |
| 9. | D8 | 9. | D7 |
| 10. | $\overline{\text{ACK}}$ | 10. | $\overline{\text{ACK}}$ |
| 11. | BUSY | 11. | BUSY |
| | | 12. | PAGE END |
| | | 13. | SELECT |
| | | 14. | AUTO END |
| 32. | $\overline{\text{ERROR}}$ | 15. | $\overline{\text{ERROR}}$ |
| | | 16. | INITIATE |
| | | 17. | SELECTION |
| 16. | SG | 18-25. | SG |



# OSBORNE

## The Connecting Cable

Use the cable available from your Gemini dealer, or you can make your own cable according to the pin configuration chart below.

## Setting the DIP Switches

Turn switches 2-3 and 2-4 to the "off" position on your Gemini, and set switch 2-1 as you like. Switch 2-2 should be set "off" if you want Gemini to print every time it receives a CR code; set it "on" if you wish to have the carriage return ignored and printing only when the buffer is full.

## The Use of BASIC

All the sample programs provided in this user's manual will run "as is" using Microsoft BASIC. If you are a first-time user, you should start with the CPM software and reset the "printer" configuration from "serial" to "Centronics." Then load MBASIC and go for it.

When you want to change the number of columns in printing, substitute

WIDTH LPRINT 132 *(or any acceptable number)*

instead of

WIDTH "LPT # 1";132 *(as used by IBM-PC).*

## Pin Configurations

The following chart shows the pin-outs for Gemini printers and Osborne, using a parallel interface.

**Parallel**

| Gemini-10X/15X | | Osborne 1 | |
|---|---|---|---|
| 2. | D1 | 1. | D0 |
| 6. | D5 | 2. | D4 |
| 3. | D2 | 3. | D1 |
| 7. | D6 | 4. | D5 |
| 4. | D3 | 5. | D2 |
| 8. | D7 | 6. | D6 |
| 5. | D4 | 7. | D3 |
| 9. | D8 | 8. | D7 |
| 1. | $\overline{\text{STROBE}}$ | 11. | $\overline{\text{STROBE}}$ |
| 11. | BUSY | 15. | BUSY |
| 16. | SG | 16. | SG |

## Program Listings

```
10 WIDTH LPRINT 255
20 E$=CHR$(27):LF$=CHR$(10):FF$=CHR$(12):CR$=CHR$(13)
30 PC$=E$+"B"+CHR$(1):EL$=E$+"B"+CHR$(2):CM$=E$+"B"+CHR$(3)
40 SP$=E$+"S"+CHR$(0):SB$=E$+"S"+CHR$(1):CS$=E$+"T"
50 IT$=E$+CHR$(52)
60 WD$=CHR$(14):CW$=CHR$(20) :PS$=E$+"@"
70 EM$=E$+"E":CE$=E$+"F":DB$=E$+"G":CD$=E$+"H"
80 LPRINT PS$;E$;"R";CHR$(15)
90 FOR I=1 TO 2
100 IF I=1 THEN LPRINT WD$;"    GEMINI STANDARD TYPE STYLES";CW$;LF$;
110 IF I=2 THEN LPRINT IT$;WD$;"     GEMINI ITALIC TYPE STYLE";
115 LPRINT CW$;LF$
120 FOR N=1 TO 40:LPRINT CHR$(231);CHR$(232);:NEXT N
130 LPRINT"          17 CPI  12 CPI  10 CPI  8.5 CPI     6 CPI        5 CPI"
140 FOR N=1 TO 80:LPRINT CHR$(241);:NEXT N
150 MS6$(1)="  ABC":MS6$(2)="X"+CHR$(120):MS6$(3)="Y"+CHR$(121)
160 FOR J=1 TO 4
170 IF I=2 THEN LPRINT IT$;
180 IF J=1 THEN LPRINT CM$;"NORMAL          ";
190 IF J=2 THEN LPRINT CM$ ;"DOUBLE STRIKE   ";
200 IF J=3 THEN LPRINT CM$ ;"EMPHASIZED      ";
210 IF J=4 THEN LPRINT CM$ ;"DOUBLE STRIKE   ";
220 FOR K=1 TO 6
230 IF K=1 THEN LPRINT CM$;
240 IF K=2 THEN LPRINT  EL$;
250 IF K=3 THEN LPRINT PC$;
260 IF K=4 THEN LPRINT CM$;WD$;
270 IF K=5 THEN LPRINT EL$;WD$;
280 IF K=6 THEN LPRINT PC$;WD$;
290 IF J=2 THEN LPRINT DB$;
300 IF J=3 THEN LPRINT EM$;
310 IF J=4 THEN LPRINT DB$;EM$;
320 LPRINT MS6$(1);SP$;MS6$(2);CS$;SB$;MS6$(3);CS$;CD$;
330 IF K>3 THEN LPRINT CW$;
340 NEXT K
350 IF J=4 THEN LPRINT E$;"A";CHR$(8);LF$;CM$;"       +";LF$;"EMPHASIZED";
360 LPRINT PS$
365 IF J=2 THEN LPRINT LF$;
```

```
370 NEXT J
380 FOR M=1 TO 5:LPRINT LF$:NEXT M
390 NEXT I
400 END
```

*Figure E-1 is the font styles program (refer to Figure 5-1) amended for the Osborne computer.*

```
5 CLR$=CHR$(26)
10 PRINT CLR$:DIM A$(22,52):X=1:Y=1
15 S$=CHR$(27)+CHR$(61)
17 DEF FNL$(X,Y)=S$+CHR$(X+32)+CHR$(Y+32)
20 PRINT CLR$:X=1:Y=1
30 GOSUB 150:PRINT FNL$(X,Y);:
40 B$=INKEY$:IF B$="" THEN 40
50 IF B$=CHR$(5) THEN PRINT CLR$:END
60 IF B$=CHR$(18) THEN PRINT CLR$:GOSUB 200:GOTO 20
70 IF B$=CHR$(16) THEN GOTO 210
80 IF B$=CHR$(13) THEN GOSUB 700:GOTO 40
90 IF B$=CHR$(4) THEN GOSUB 130:GOTO 40
100 PRINT FNL$(X,Y);:PRINT B$;:A$(X,Y)=B$:Y=Y+1
110 IF Y=50 THEN A$(X,Y)=CHR$(13):PRINT CHR$(7);:X=X+1:Y=1:GOTO 40
120 PRINT FNL$(X,Y);:PRINT "";:GOTO 40
130 PRINT FNL$(X,Y);:PRINT " ";:Y=Y-1:IF Y=0 THEN Y=1:PRINT CHR$(7);
135 A$(X,Y)=""
140 PRINT FNL$(X,Y);:PRINT "";:RETURN
150 PRINT FNL$(22,2);
160 PRINT "^P)RINTER  ^D)ELETE  ^R)ESTART  ^E)XIT";
190 RETURN
200 PRINT FNL$(10,20);"PLEASE STAND BY";
205 FOR I=1 TO X:FOR J=1 TO 50:A$(I,J)="":
206 NEXT J:NEXT I:RETURN
210 REM
220 LPRINT CHR$(27);"@"
230 PRINT CLR$ :PRINT FNL$(23,2);
235 PRINT "C)OMMAND";:PRINT FNL$(23,11);
240 PRINT "P)RINT";FNL$(23,18);"R)ETURN";
250 G$=INKEY$:IF G$="" THEN 250
260 IF G$="P" THEN GOSUB 310
```

```
270 IF G$="C" THEN GOSUB 320:GOTO 230
280 IF G$="R" THEN GOSUB 300:GOTO 30
290 GOTO 250
300 PRINT CLR$:FOR I=1 TO X:FOR J=1 TO 50
302 PRINT FNL$(I,J);:
304 IF A$(I,J)=CHR$(13) THEN 308
306 PRINT A$(I,J);:
308 NEXT J:NEXT I
309 RETURN
310 FOR I=1 TO X:FOR J=1 TO 50:LPRINT A$(I,J);
312 NEXT J:LPRINT :NEXT I:RETURN
320 PRINT FNL$(23,1);SPC(47);
340 PRINT FNL$(23,2);" E) Esc";
350 PRINT FNL$(23,10);" C) CHR$ ";
355 PRINT FNL$(23,20);" Q) END COMMAND ";:ST$=""
360 G$=INKEY$:IF G$="" THEN 360
370 IF G$="E" THEN GOSUB 420:GOTO 400
380 IF G$="C" THEN GOSUB 530:GOTO 400
390 IF G$="Q" THEN GOTO 410
400 GOTO 360
410 RETURN
420 I=0:PRINT FNL$(20,4);:PRINT "Esc ";
430 G$=INKEY$:IF G$="" THEN 430
440 IF G$=CHR$(13) THEN 480
450 PRINT G$;
460 PRINT " ";
470 S$(I)=G$:I=I+1:GOTO 430
480 IF I>2 THEN GOSUB 800
490 IF I=2 THEN GOSUB 850
500 IF I=1 THEN ST$=CHR$(27)+S$(0)
510 LPRINT ST$:ST$="":FOR J=1 TO I:S$(J)=""
512 NEXT J:I=0
520 PRINT FNL$(20,4);:PRINT STRING$(20," ")
521 RETURN
530 PRINT FNL$(20,4);" CHR$ ";:ST$=""
540 G$=INKEY$:IF G$="" THEN 540
550 IF G$=CHR$(13) THEN 570
560 PRINT G$;:ST$=ST$+G$:GOTO 540
570 PRINT FNL$(20,4);:PRINT STRING$(10," ")
580 LPRINT CHR$(VAL(ST$));:ST$="":RETURN
700 REM
```

```
710 A$(X,Y)=B$
720 PRINT FNL$(X,Y);" ";
730 X=X+1:Y=1:
740 PRINT FNL$(X,Y);"";:RETURN
800 FOR J=1 TO I:D$=D$+S$(J):NEXT J:
810 ST$=CHR$(27)+S$(0)+CHR$(VAL(D$)):D$=""
820 RETURN
850 ST$=CHR$(27)+S$(0)+CHR$(VAL(S$(1)))
860 RETURN
```

*Figure E-2 is the word processing program (refer to Figure 5-4) amended for the Osborne computer.*

```
10 LPRINT TAB(30);"HIGH RESOLUTION VERTICAL SPACING"
20 LPRINT TAB(30);"USING THE ESC 3 N  CONTROL CODE"
30 FOR I=1 TO 30
40 IF I=13 THEN GOTO 130
50 M=INT(144/(6*I)):IF M<2 THEN M=2
60 LPRINT CHR$(27);CHR$(51);CHR$(I)
70 FOR J=0 TO M
80 IF J=M-1 THEN LPRINT I;"/144 INCH SPACING          ";:GOTO 100
90 LPRINT TAB(30);
100 FOR N=1 TO 40:LPRINT CHR$(241);:NEXT N:LPRINT  " "
110 NEXT J
130 NEXT I
140 LPRINT CHR$(27);"@"
150 END
```

*Figure E-3 is the vertical spacing program (refer to Figure 6-5) amended for the Osborne computer.*

```
10 LPRINT CHR$(27);"@";CHR$(27);"A";CHR$(6)
20 FOR LI=1 TO 2
30 LPRINT CHR$(27);"K";CHR$(114);CHR$(0);
40 FOR SPACE=1 TO 100
50 LPRINT CHR$(0);:NEXT SPACE
60 FOR COLM=1 TO 14
70 READ C:LPRINT CHR$(C);:NEXT COLM
80 LPRINT :NEXT LI
90 DATA 14,16,34,64,140,140,128,128,140,140,64,34,16,14
100 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
110 LPRINT CHR$(27);"@":
120 LPRINT :LPRINT :LPRINT
130 END
```

*Figure E-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the Osborne computer.*

```
10 LPRINT CHR$(163);CHR$(161);
20 LPRINT CHR$(27);"A";CHR$(3)
30 LPRINT CHR$(162);CHR$(160);
40 LPRINT CHR$(27);"2"
50 END
```

*Figure E-5 is the circle program presented in Chapter 7, amended for the Osborne computer.*

```
10 LPRINT TAB(2);CHR$(163);CHR$(27);"A";CHR$(6)
20 LPRINT CHR$(160)
30 END
```

*Figure E-6 is the tilde program presented in Chapter 7 amended for the Osborne computer.*

```
10 LPRINT TAB(20);"N";CHR$(13)
20 LPRINT TAB(20);CHR$(164);CHR$(13)
30 LPRINT TAB(18);"W";CHR$(166);CHR$(32);CHR$(167);"E";CHR$(13)
40 LPRINT TAB(20);CHR$(165);CHR$(13)
50 LPRINT TAB(20);"S"
60 END
```

*Figure E-7 is the map program presented in Chapter 7 amended for the Osborne computer.*

```
15 LPRINT CHR$(27);"U";CHR$(1)
20 LPRINT CHR$(27);"A";CHR$(6)
30 FOR J=1 TO 9
40 FOR I=1 TO 12
50 READ A
60 LPRINT CHR$(A);
70 NEXT I
80 LPRINT
90 NEXT J
100 LPRINT CHR$(27);"@":
120 END
130 DATA 239,224,224,224,239,224,224,224,224,224,224,224
140 DATA 239,224,224,224,239,224,224,224,224,224,224,224
150 DATA 239,224,224,224,239,224,239,224,224,224,239,224
160 DATA 239,224,239,224,239,224,234,233,224,234,233,224
170 DATA 239,234,239,233,239,224,224,239,224,239,224,224
180 DATA 239,239,224,239,239,224,224,234,239,233,224,224
190 DATA 234,233,224,234,233,224,224,224,239,224,224,224
200 DATA 224,224,224,224,224,224,224,234,233,224,224,224
210 DATA 224,224,224,224,224,224,224,239,224,224,224,224
```

*Figure E-8 is the "Wy" program (refer to Figure 7-1) amended for the Osborne computer.*

```
10 LPRINT CHR$(27);"A";CHR$(6);CHR$(15)
20 LPRINT CHR$(27);"U";CHR$(1)
30 FOR K=1 TO 3
40 LPRINT CHR$(27);"B";CHR$(K)
50 RESTORE
60 FOR J=1 TO 9
70 FOR I=1 TO 12
80 READ A:LPRINT CHR$(A);
90 NEXT I
100 LPRINT
110 NEXT J
120 LPRINT:NEXT K
130 LPRINT CHR$(27);"@":END
140 DATA 224,224,224,224,224,224,224,224,224,245,224,224
150 DATA 224,224,224,224,224,224,224,224,252,245,224,224
160 DATA 224,224,224,245,224,224,224,252,239,245,224,224
```

```
170 DATA 224,224,252,245,224,224,252,239,239,245,224,224
180 DATA 224,252,239,245,224,252,239,239,239,245,224,224
190 DATA 252,239,239,245,252,239,239,239,239,245,224,224
200 DATA 241,241,241,249,241,241,241,241,241,249,224,224
210 DATA 239,239,239,239,239,239,239,239,239,239,239,251
220 DATA 253,239,239,239,239,239,239,239,239,239,251,224
```

*Figure E-9 is the boat program (refer to Figure 7-2) amended for the Osborne computer.*

```
10 PRINT CHR$(26)
20 PRINT "":PRINT "":PRINT ""
30 PRINT "THIS PROGRAM TAKES ABOUT TWO"
40 PRINT "MINUTES TO RUN, PLEASE TURN"
50 PRINT "ON YOUR PRINTER AND STAND BY"
100 REM MICRO-PLOT
110 DIM BIT%(75,12)
1000 REM SET PROGRAM CONSTANTS
1010 MASK%(1)=128:MASK%(4)=16
1020 MASK%(2)=64 :MASK%(5)=8
1030 MASK%(3)=32 :MASK%(6)=4
1040 LX=20       :LY=20
1050 XFAC=72/LX  :YFAC=75/LY
2000 REM PLOT CURVE
2010 RAD=9
2020 X1=19  :Y1=10
2030 FOR ANG%=0 TO 360 STEP 10
2040 R1=ANG%*6.28/360
2050 X2=RAD*COS(R1)+10 :Y2=RAD*SIN(R1)+10
2060 GOSUB 4000
2070 NEXT ANG%
2080 FOR ANG%=0 TO 360 STEP 15
2090 R1=ANG%*6.28/360
2100 R2=(ANG%+150)*6.28/360
2110 X1=RAD*COS(R1)+10  :Y1=RAD*SIN(R1)+10
2120 X2=RAD*COS(R2)+10  :Y2=RAD*SIN(R2)+10
2130 GOSUB 4000
2140 NEXT ANG%
3000 REM SEND BIT IMAGE MAP TO PRINTER
3010 LPRINT CHR$(27);"A";CHR$(6)
3020 FOR ROW%=0 TO 11
```

```
3030 A$=""
3040 LPRINT CHR$(27);"K";CHR$(119);CHR$(1);
3050 FOR COL% = 1 TO 75
3055 FOR LL=1 TO 5
3060 LPRINT CHR$(BIT%(COL%,ROW%));
3065 NEXT LL
3070 NEXT COL%
3080 LPRINT
3090 NEXT ROW%
3100 LPRINT CHR$(27);"A";CHR$(12)
3110 END
4000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
4010 XL=X2-X1 :YL=Y2-Y1
4020 NX=ABS(XL!XFAC) : NY=ABS(YL!YFAC)
4030 IF NX<NY THEN NX=NY
4040 NS%=INT(NX+1)
4050 DX=XL/NS%  : DY=YL/NS%
4060 FOR I%=1 TO NS%
4070 X1=X1+DX   :  Y1=Y1+DY
4080 GOSUB 5000
4090 NEXT I%
4100 RETURN
5000 REM PLOT A POINT AT X1,Y1
5010 XX=X1!XFAC   :YY=Y1!YFAC
5020 COL%=INT(XX)+1
5030 ROW%=INT(YY/6)
5040 XIT%=INT(YY-ROW%!6)+1
5050 BIT%(COL%,ROW%)=BIT%(COL%,ROW%) OR MASK%(XIT%)
5060 RETURN
```

*Figure E-10 is the graphics program (refer to Figure 7-4) amended for the Osborne computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE PARAGRAPH SYMBOL IN PLACE OF @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(6);
45 LPRINT CHR$(25);CHR$(0);CHR$(25);CHR$(118);CHR$(25);CHR$(0);
47 LPRINT CHR$(127);CHR$(0)
50 REM SELECT DOWNLOADABLE CHARACTERS
```

```
60 LPRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "@@@@@@@@"
90 END
```

*Figure E-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the Osborne computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 LPRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);
47 LPRINT CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "COPY RIGHT @ 1983"
90 END
```

*Figure E-12 is the downloading copyright program (refer to Figure 8-4) amended for the Osborne computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
24 REM COPY CHARACTERS TO ROM
25 LPRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 LPRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);
47 LPRINT CHR$(34);CHR$(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "COPY RIGHT @ 1983"
90 END
```

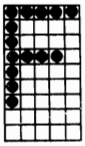
*Figure E-13 is the downloading program presented in Figure 8-5 amended for the Osborne computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
24 REM COPY CHARACTERS TO ROM
25 LPRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);
45 LPRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);
47 LPRINT CHR$(34);CHR$(20)
49 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF &
50 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(38);CHR$(1);CHR$(28);
55 LPRINT CHR$(34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);
57 LPRINT CHR$(65);CHR$(34);CHR$(20)
59 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "@&@&@&@&@&@&@&@&"
90 END
```

*Figure E-14 is the downloading program presented in Figure 8-6 amended for the Osborne computer.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 LPRINT ESC$;CHR$(43);ESC$;"E";ESC$;"W";CHR$(1);CHR$(30)
40 REM ISSUE MACRO COMMAND
50 LPRINT ESC$;CHR$(33)
60 LPRINT "EMPHASIZED AND EXPANDED BY MACRO"
70 END
```

*Figure E-15 is the macro-commands program (refer to Figure 8l-7) amended for the Osborne computer.*



# TRS-80

## The Connecting Cable

Use the cable supplied by your Gemini dealer for your TRS Model II or Model III computer. No interface kit is required for these models.

## Setting the DIP Switches

Set switch number 2-3 to the "off" position on your Gemini. You may set switch number 2-1 in either position, at your option.

Switch number 2-2 should be set "off" if you want Gemini to print every time it receives a CR code; set to "on" if you wish to have the carriage return ignored and printing only when the buffer is full.

The TRS-80 computer does not automatically send a line feed at the end of a line. If you want to insert a line feed (when, for example, listing a program), turn DIP switch number 2-4 to the "on" position.

## The Use of BASIC

The Model II computer may need to be initialized to properly run the sample programs in this user's manual. From BASIC, type:

```
SYSTEM "FORMS"
```

and respond to the questions asked by the computer. Then, type:

```
Q
```

and the system will correctly route LPRINT commands to your Gemini.

The TRS-80 Microsoft BASIC is quite similar to the Microsoft BASIC used in the demonstration programs. The only exception is the width commands, which are explained in the documentation accompanying the TRS-80 Microsoft BASIC programs. You may also refer to the amended program listing at the back of this Appendix.

## Pin Configurations

The following chart shows the pin-outs for Gemini printers and the TRS-80 Models I and II with expansion interface.

**Parallel**

| Gemini-10X/15X | | Model I | |
|---|---|---|---|
| 1. | $\overline{\text{STROBE}}$ | 1. | $\overline{\text{STROBE}}$ |
| 2. | D1 | 3. | D1 |
| 3. | D2 | 5. | D2 |
| 4. | D3 | 7. | D3 |
| 5. | D4 | 9. | D4 |
| 6. | D5 | 11. | D5 |
| 7. | D6 | 13. | D6 |
| 8. | D7 | 15. | D7 |
| 9. | D8 | 17. | D8 |
| 11. | BUSY | 21. | READY |

**Parallel**

| Gemini-10X/15X | | Model II | |
|---|---|---|---|
| 1. | $\overline{\text{STROBE}}$ | 1. | $\overline{\text{STROBE}}$ |
| 2. | D1 | 3. | D1 |
| 3. | D2 | 5. | D2 |
| 4. | D3 | 7. | D3 |
| 5. | D4 | 9. | D4 |
| 6. | D5 | 11. | D5 |
| 7. | D6 | 13. | D6 |
| 8. | D7 | 15. | D7 |
| 9. | D8 | 17. | D8 |
| 10. | $\overline{\text{ACK}}$ | 19. | $\overline{\text{ACK}}$ |
| 11. | BUSY | 21. | BUSY |

## Program Listings

```
10 POKE 16427,255
20 E$=CHR$(27):LF$=CHR$(10):FF$=CHR$(12):CR$=CHR$(13)
30 PC$=E$+"B"+CHR$(1):EL$=E$+"B"+CHR$(2):CM$=E$+"B"+CHR$(3)
40 SP$=E$+"S"+CHR$(0):SB$=E$+"S"+CHR$(1):CS$=E$+"T"
50 IT$=E$+CHR$(52)
60 WD$=CHR$(14):CW$=CHR$(20):PS$=E$+"@"
70 EM$=E$+"E":CE$=E$+"F":DB$=E$+"G":CD$=E$+"H"
80 LPRINT PS$;E$;"R";CHR$(15);
90 FOR I=1 TO 2
100 IF I=1 THEN LPRINT WD$;"   GEMINI STANDARD TYPE STYLES";CW
$;LF$
110 IF I=2 THEN LPRINT IT$;WD$;"    GEMINI ITALIC TYPE STYLE";
CW$;LF$
120 FOR N=1 TO 40:LPRINT CHR$(231);CHR$(232);:NEXT N:LPRINT:LP
RINT
130 LPRINT "       17 CPI  12 CPI  10 CPI    8.5 CPI    6 CPI
     5 CPI"
140 FOR N=1 TO 80:LPRINT CHR$(241);:NEXT N:LPRINT
150 M1$=" ABC":M2$="X"+CHR$(120):M3$="Y"+CHR$(121)
160 FOR J=1 TO 4
170 IF I=2 THEN LPRINT IT$;
180 IF J=1 THEN LPRINT CM$;"NORMAL        ";
190 IF J=2 THEN LPRINT CM$;"DOUBLE STRIKE ";
200 IF J=3 THEN LPRINT CM$;"EMPHASIZED    ";
210 IF J=4 THEN LPRINT CM$;"DOUBLE STRIKE ";
220 FOR K=1 TO 6
230 IF K=1 THEN LPRINT CM$;
240 IF K=2 THEN LPRINT EL$;
250 IF K=3 THEN LPRINT PC$;
260 IF K=4 THEN LPRINT CM$;WD$;
270 IF K=5 THEN LPRINT EL$;WD$;
280 IF K=6 THEN LPRINT PC$;WD$;
290 IF J=2 THEN LPRINT DB$;
300 IF J=3 THEN LPRINT EM$;
310 IF J=4 THEN LPRINT DB$;EM$;
320 LPRINT M1$;SP$;M2$;CS$;SB$;M3$;CS$;CD$;
330 IF K>3 THEN LPRINT CW$;
340 NEXT K
```

```
350 IF J=4 THEN LPRINT E$;"A";CHR$(8);LF$;CM$;"     + ";CR$;"E
MPHASIZED";
360 LPRINT PS$
370 IF J=2 THEN LPRINT LF$;
375 NEXT J
380 FOR M=1 TO 3:LPRINT :NEXT M
390 NEXT I
400 END
```

*Figure F-1 is the font styles program (refer to Figure 5-1) amended for the TRS-80.*

```
5 CLEAR 200
7 C$=CHR$(179)
10 CLS:DIM A$(16,62)
20 CLS:X=0:Y=0
30 GOSUB 150:PRINT @X*64+Y,C$;
40 B$=INKEY$:IF B$="" THEN 40
50 IF B$=CHR$(64) THEN CLS:END
60 IF B$=CHR$(31) THEN CLS:GOSUB 200:GOTO 20
70 IF B$=CHR$(9) THEN GOTO 210
80 IF B$=CHR$(13) THEN A$(X,Y)=B$:PRINT @X*64+Y," ":X=X+1:Y=0:PRINT @X*64+Y,C$;:GOTO 40
90 IF B$=CHR$(8) THEN GOSUB 130:GOTO 40
100 PRINT @X*64+Y,B$;:A$(X,Y)=B$:Y=Y+1:
110 IF Y=62 THEN A$(X,Y)=CHR$(13):X=X+1:Y=0:GOTO 40
120 PRINT @X*64+Y,C$;:GOTO 40
130 PRINT @X*64+Y," ";:Y=Y-1:IF Y<0 THEN Y=0:
140 A$(X,Y)="":PRINT @X*64+Y,C$;:RETURN
150 PRINT @15*64+0," <-- BACK SPCAE    --> PRINT   <CLEAR>RESTART   @ EXIT";:RETURN
200 PRINT @7*64+20,"PLEASE STAND BY";
205 FOR I=0 TO X:FOR J=0 TO 62:A$(I,J)="":NEXT J:NEXT I
206 RETURN
210 REM
220 LPRINT CHR$(27);"@"
230 CLS:PRINT @15*64+2,"C)OMMAND  P)RINT   R)ETURN";
250 G$=INKEY$:IF G$="" THEN 250
260 IF G$="P" THEN GOSUB 310
270 IF G$="C" THEN GOSUB 320:GOTO 230
280 IF G$="R" THEN GOSUB 300:GOTO 30
290 GOTO 250
300 CLS:FOR I=0 TO X:FOR J=0 TO 62
```

```
302 PRINT @I%64+J,A$(I,J);:NEXT J:NEXT I:RETURN
310 FOR I=0 TO X:FOR J=0 TO 62:LPRINT A$(I,J);
312 NEXT J:NEXT I:RETURN
320 PRINT @15%64+2,STRING$(50," ");
330 PRINT @15%64+2," E) ESC   C) CHR$   Q) END COMMAND ";
340 ST$=""
360 G$=INKEY$:IF G$="" THEN 360
370 IF G$="E" THEN GOSUB 420:GOTO 400
380 IF G$="C" THEN GOSUB 530:GOTO 400
390 IF G$="Q" THEN GOTO 410
400 GOTO 360
410 RETURN
420 I=0:PRINT @13%64+3,"Esc ";
430 G$=INKEY$:IF G$="" THEN 430
440 IF G$=CHR$(13) THEN 480
450 PRINT G$;" ";
470 S$(I)=G$:I=I+1:GOTO 430
480 IF I>2 THEN FOR J=1 TO I:D$=D$+S$(J):NEXT J:ST$=CHR$(27)+S$(0)+CHR$(VAL(D$)):D$="":
490 IF I=2 THEN ST$=CHR$(27)+S$(0)+CHR$(VAL(S$(1))):
500 IF I=1 THEN ST$=CHR$(27)+S$(0)
510 LPRINT ST$:ST$="":FOR J=1 TO I:S$(J)="":NEXT J:I=0
520 PRINT @13%64+2,STRING$(20," ");:RETURN
530 PRINT @13%64+3," CHR$ ";:ST$=""
540 G$=INKEY$:IF G$="" THEN 540
550 IF G$=CHR$(13) THEN 570
560 PRINT G$;:ST$=ST$+G$:GOTO 540
570 PRINT @13%64+2,STRING$(20," ");
580 LPRINT CHR$(VAL(ST$));:ST$="":RETURN
```

*Figure F-2 is the word processing program (refer to Figure 5-4) amended for the TRS-80.*

```
10 LPRINT TAB(30);"HIGH RESOLUTION VERTICAL SPACING "
15 LPRINT TAB(30);"USING THE ESC  3  N CONTROL CODE "
20 FOR I = 1 TO 30
25 IF I=12 THEN 110
30 IF I = 13 THEN GOTO 110
40 N = INT(144/(6%I)): IF N < 2 THEN N = 2
50 LPRINT CHR$(27);CHR$(51);CHR$(I);
60 FOR J = 0 TO N
```

```
70 IF J = M-1 THEN LPRINT I;"/144 INCH SPACING          ";:GOT
O 90
80 LPRINT TAB(30);
90 FOR N = 1 TO 40 : LPRINT CHR$(241); : NEXT N:LPRINT
100 NEXT J
110 NEXT I
120 LPRINT CHR$(27);"@":END
```

*Figure F-3 is the vertical spacing program (refer to Figure 6-5) amended for the TRS-80.*

```
10 RESTORE
20 LPRINT CHR$(27);"@";CHR$(27);"1"
30 FOR LI = 1 TO 2
40 LPRINT CHR$(27);"K";CHR$(114);CHR$(0);
50 FOR SPACE = 1 TO 100:LPRINT CHR$(0);:NEXT SPACE
60 FOR COLM = 1 TO 14:READ C:LPRINT CHR$(C);:NEXT COLM
70 LPRINT : NEXT LI
80 DATA 14,16,34,64,140,140,128,128,140,140,64,34,16,14
90 DATA 224,16,232,68,34,18,18,18,18,34,68,232,16,224
100 LPRINT CHR$(27);"@":LPRINT:LPRINT:LPRINT
```

*Figure F-4 is the dot graphics program presented in Chapter 6 (the one that puts a smile on your paper) amended for the TRS-80.*

```
10 LPRINT CHR$(163);CHR$(161);CHR$(27);"A";CHR$(6)
20 LPRINT CHR$(162);CHR$(160);CHR$(27);"2"
30 END
```

*Figure F-5 is the circle program presented in Chapter 7, amended for the TRS-80.*

```
10 LPRINT TAB(1);CHR$(163);CHR$(27);"A";CHR$(6)
20 LPRINT CHR$(160)
30 END
```

*Figure F-6 is the tilde program presented in Chapter 7 amended for the TRS-80.*

```
10 LPRINT TAB(20);"N";CHR$(13);TAB(20);CHR$(164);CHR$(13);TAB(
16);"W ";CHR$(166);CHR$(32);TAB(22);CHR$(167);" E";CHR$(13);TA
B(20);CHR$(165);CHR$(13);TAB(20);"S"
15 END
```

*Figure F-7 is the map program presented in Chapter 7 amended for the TRS-80.*

```
10 LPRINT CHR$(27);"A";CHR$(6)
15 LPRINT CHR$(27);"U";CHR$(1)
20 FOR J = 1 TO 9
30 FOR I = 1 TO 12
40 READ A
50 LPRINT CHR$(A);
60 NEXT I
70 LPRINT
80 NEXT J
90 DATA 239,224,224,224,239,224,  224,224,224,224,224,224
91 DATA 239,224,224,224,239,224,  224,224,224,224,224,224
92 DATA 239,224,224,224,239,224,  239,224,224,224,239,224
93 DATA 239,224,239,224,239,224,  234,233,224,234,233,224
94 DATA 239,234,239,233,239,224,  224,239,224,239,224,224
95 DATA 239,239,224,239,239,224,  224,234,239,233,224,224
96 DATA 234,233,224,234,233,224,  224,224,239,224,224,224
97 DATA 224,224,224,224,224,224,  224,234,233,224,224,224
98 DATA 224,224,224,224,224,224,  224,239,224,224,224,224
100 LPRINT CHR$(27);"@":END
```

*Figure F-8 is the "Wy" program (refer to Figure 7-1) amended for the TRS-80.*

```
5 LPRINT CHR$(27);"U";CHR$(1)
10 LPRINT CHR$(27);"A";CHR$(6);CHR$(15)
15 FOR K = 1 TO 3
16 LPRINT CHR$(27);"B";CHR$(K)
17 RESTORE
20 FOR J = 1 TO 9
30 FOR I = 1 TO 12
40 READ A
50 LPRINT CHR$(A);
```

```
60 NEXT I
70 LPRINT
80 NEXT J
90 DATA 224,224,224,224,224,224,  224,224,224,245,224,224
91 DATA 224,224,224,224,224,224,  224,224,252,245,224,224
92 DATA 224,224,224,245,224,224,  224,252,239,245,224,224
93 DATA 224,224,252,245,224,224,  252,239,239,245,224,224
94 DATA 224,252,239,245,224,252,  239,239,239,245,224,224
95 DATA 252,239,239,245,252,239,  239,239,239,245,224,224
96 DATA 241,241,241,249,241,241,  241,241,241,249,224,224
97 DATA 239,239,239,239,239,239,  239,239,239,239,239,251
98 DATA 253,239,239,239,239,239,  239,239,239,239,251,224
99 LPRINT :NEXT K
100 LPRINT CHR$(27);"@":END
```

*Figure F-9 is the boat program (refer to Figure 7-2) amended for the TRS-80.*

```
10 CLS:PRINT "":PRINT "":PRINT ""
20 PRINT "THIS PROGRAM TAKES ABOUT 4 MINUTES TO RUN"
30 PRINT "PLEASE TURN ON YOUR PRINTER AND STAND  BY"
100 DIM BIT%(76,12)
110 MASK%(1)=128:MASK%(4)=16
120 MASK%(2)=64 :MASK%(5)=8
130 MASK%(3)=32 :MASK%(6)=4
140 LX=20:LY=20
150 XFAC=72/LX:YFAC=75/LY
1000 REM PLOT CURVE
1010 RAD=9
1020 X1=19:Y1=10
1030 FOR ANG=0 TO 360 STEP 10
1040 R1=ANG*6.28/360
1050 X2=RAD*COS(R1)+10:Y2=RAD*SIN(R1)+10
1060 GOSUB 3000
1070 NEXT
1080 FOR ANG=0 TO 360 STEP 15
1090 R1=ANG*6.28/360
1100 R2=(ANG+150)*6.28/360
1110 X1=RAD*COS(R1)+10:Y1=RAD*SIN(R1)+10
1120 X2=RAD*COS(R2)+10:Y2=RAD*SIN(R2)+10
```

```
1130 GOSUB 3000
1140 NEXT
2000 GOSUB 5000
2010 LPRINT CHR$(27);"A";CHR$(6)
2020 FOR ROW=0 TO 11
2040 LPRINT CHR$(27);"K";CHR$(119);CHR$(1);
2050 FOR COL=1 TO 75
2060 FOR LL=1 TO 5
2070 LPRINT CHR$(BIT%(COL,ROW));
2080 NEXT:NEXT:LPRINT:NEXT:LPRINT
2090 END
3000 REM DRAW A LINE FROM X1,Y1 TO X2,Y2
3010 XL=X2-X1:YL=Y2-Y1
3020 NX=ABS(XL*XFAC):NY=ABS(YL*YFAC)
3030 IF NX<NY THEN NX=NY
3040 NS%=INT(NX+1)
3050 DX=XL/NS%:DY=YL/NS%
3060 FOR I=1 TO NS%
3070 X1=X1+DX:Y1=Y1+DY
3080 GOSUB 4000
3090 NEXT I
3100 RETURN
4000 REM PLOT A POINT AT X1,Y1
4010 XX=X1*XFAC:YY=Y1*YFAC
4020 COL=INT(XX+1)
4030 ROW=INT(YY/6)
4040 XIT%=INT(YY-(6*ROW))+1
4050 BIT%(COL,ROW)=BIT%(COL,ROW) OR MASK%(XIT%)
4060 RETURN
5000 FOR I=0 TO 11:FOR J=1 TO 75
5010 IF BIT%(J,I)=12 THEN BIT%(J,I)=14
5020 NEXT:NEXT:RETURN
```

*Figure F-10 is the graphics program (refer to Figure 7-4) amended for the TRS-80.*

```
10 REM DEFINE THE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE PARAGRAPH SYMBOL IN PLACE OF @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(6);CHR$(9);CHR$(0);
```

```
   CHR$(9);CHR$(118);CHR$(9);CHR$(0);CHR$(127);CHR$(0)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1)
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "@@@@@@"
90 END
```

*Figure F-11 is the downloading paragraph symbols program (refer to Figure 8-2) amended for the TRS-80.*

```
10 REM DEFINE THE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);CHR$
   (34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR
   $(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1);
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "COPY RIGHT @ 1983"
90 END
```

*Figure F-12 is the downloading copyright program (refer to Figure 8-4) amended for the TRS-80.*

```
10 REM DEFINE THE ESCAPE CHARACTER
20 ESC$=CHR$(27)
24 REM COPY CHARACTERS TO RAM
25 LPRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);CHR$
   (34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR
   $(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1);
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "COPY RIGHT @ 1983"
90 END
```

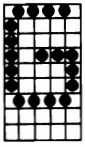
*Figure F-13 is the downloading program presented in Figure 8-5 amended for the TRS-80.*

```
10 REM DEFINE THE ESCAPE CHARACTER
20 ESC$=CHR$(27)
24 REM COPY CHARACTERS TO RAM
25 LPRINT ESC$;CHR$(42);CHR$(0)
30 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF @
40 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(64);CHR$(0);CHR$(28);CHR$
   (34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR
   $(20)
44 REM DEFINE COPYRIGHT SYMBOL IN PLACE OF &
45 LPRINT ESC$;CHR$(42);CHR$(1);CHR$(38);CHR$(1);CHR$(28);CHR$
   (34);CHR$(73);CHR$(20);CHR$(65);CHR$(20);CHR$(65);CHR$(34);CHR
   $(20)
50 REM SELECT DOWNLOADABLE CHARACTERS
60 LPRINT ESC$;CHR$(36);CHR$(1);
70 REM PRINT OUR NEW SYMBOL
80 LPRINT "@&@&@&@&@&@&@&@&@&"
90 END
```

*Figure F-14 is the downloading program presented in Figure 8-6 amended for the TRS-80.*

```
10 REM DEFINE ESCAPE CHARACTER
20 ESC$=CHR$(27)
30 REM DEFINE MACRO TO BE EMPHASIZED AND EXPANDED TEXT
40 LPRINT ESC$;CHR$(43);ESC$;"E";ESC$;"W";CHR$(1);CHR$(30)
50 REM ISSUE MACRO COMMAND AND TRY PRINTING
60 LPRINT ESC$;CHR$(33)
70 LPRINT "EMPHASIZED AND EXPANDED BY MACRO"
80 END
```

*Figure F-15 is the macro-commands program (refer to Figure 8-7) amended for the TRS-80.*



# WHEN YOU FIRST GET YOUR NEW GEMINI

This appendix serves the needs of Gemini's first-time users. On the following pages, you will learn what to look for when initially opening the box containing your new printer. In addition to learning what the pieces are, you'll gain instruction in how the pieces fit together. Specifically, this appendix will show you how to remove the packing and attach the paper-handling accessories.

## CHECKLIST FOR FIRST OPENING THE BOX

When you first open the shipping/purchase carton for your Gemini printer, you should find the following items:

### CHECKLIST FOR YOUR NEW GEMINI

- ☐ one Gemini printer
- ☐ one paper guide (a smaller metal rack)
- ☐ one paper separator (a metal rack)
- ☐ one roll-paper holder (a formed metal bar)
- ☐ one holder shaft (a long plastic cylinder)
- ☐ one ink ribbon (mounted on two spools)
- ☐ one spare fuse (small, glass & metal)
- ☐ one user's manual (this one, right here)
- ☐ a warranty card (be sure to mail it in)

## WHAT THE PARTS LOOK LIKE

*Figure G-1 shows these items for the Gemini-10X, and Figure G-2 presents them for the Gemini-15X.*

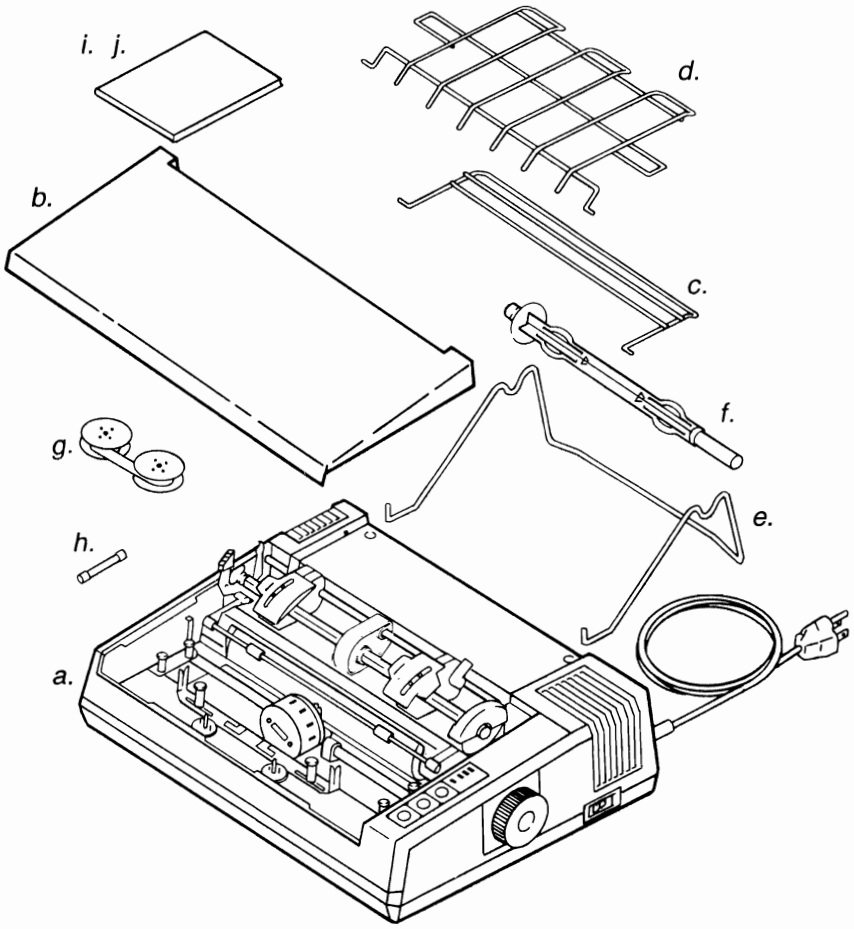



*Figure G-1. The Gemini-10X packing list.*

| | | |
|---|---|---|
| *Legend:* | *a. printer* | *f. holder shaft* |
| | *b. printer cover* | *g. ink ribbon* |
| | *c. paper guide* | *h. spare fuse* |
| | *d. paper separator* | *i. user's manual* |
| | *e. roll paper holder* | *j. warranty* |

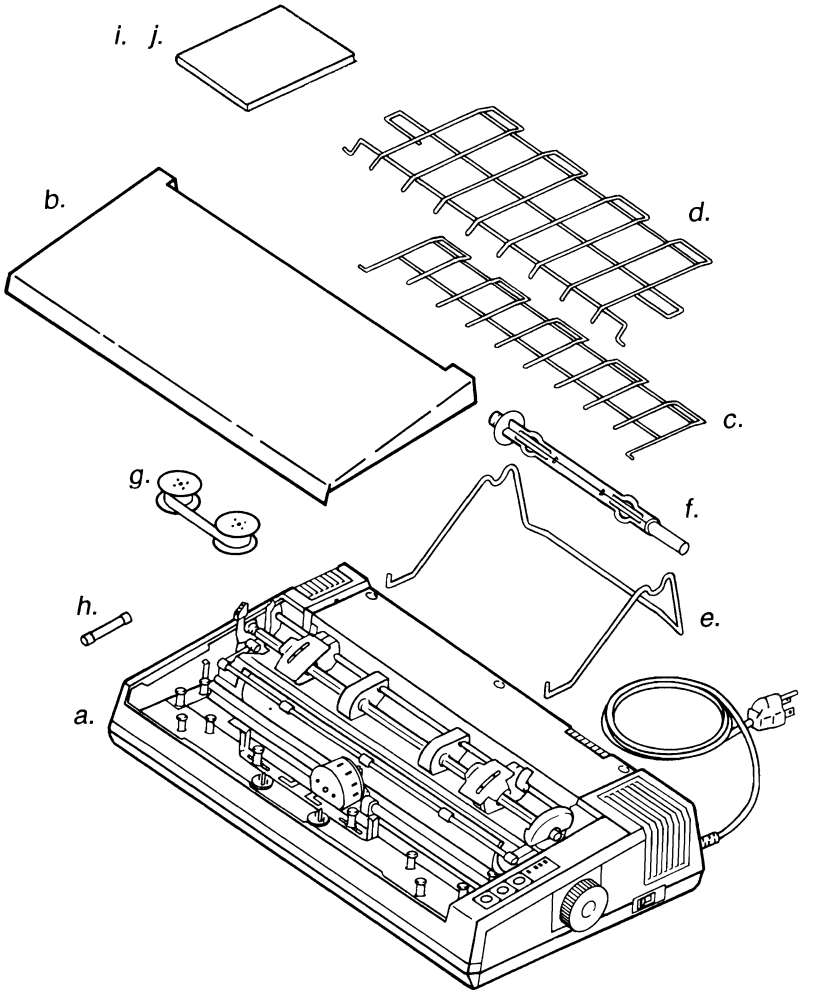



*Figure G-2. The Gemini-15X packing list.*

## REMOVING THE PACKING FROM INSIDE THE PRINTER

Your Gemini came to you nestled in foam plastic cushions. Packing material was also placed inside your printer to secure the print head and platen during shipping. Figure G-3 identifies the packing material that you need to remove from inside Gemini prior to operation.

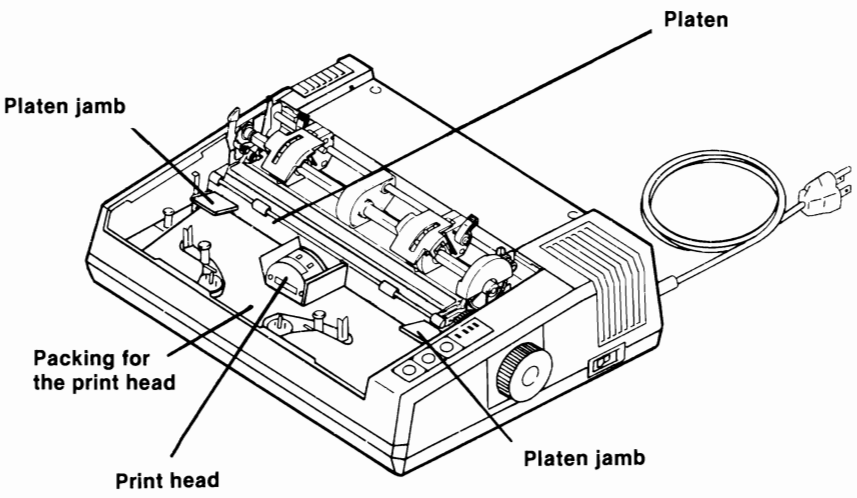



*Figure G-3. The Gemini-10X (shown here) and the Gemini-15X both use a cardboard form to secure the print head, cardboard inserts to secure the platen from jostling around.*

**Note:** Be sure you remove all the cardboard packing, before using your Gemini.

First, you need to remove the cardboard packing shown in Figure G-3.

***Note:*** You might want to save these packing materials, in case you move, or in case you need to ship your Gemini printer.

After you remove the platen-securing packing, while your printer is turned "off," turn the platen knob to roll out the sheet of paper that is wrapped around the platen (to protect the "out of paper" sensor).

## REMOVAL OF THE "FIXING" SCREWS

When your new printer arrives, there are two screws that "fix" the internal chassis to the external frame. This is similar to "fixing" screws on many phonographic turntables. You need to remove these screws to allow the chassis to "float" on its internal shock absorbers made of rubber-like soft plastic.

To remove the fixing screws, you first need to get to them. Turn your printer over and set it on a pillow or fabric-covered surface. (This is to keep from "skinning" the top of your printer while you make this quick adjustment.)

As shown in Figure G-4, remove the two packing screws. Figure G-4 illustrates the bottom of the Gemini-15X, which has a slot for feeding sprocket paper through the printer's underneath (called "bottom pass"). If you have a Gemini-10X the underside of your printer will not look exactly like the Gemini-15X but there are only two screws that can be removed, so you can't go wrong.

***Note:*** You should save the screws, too. It's a good idea to tape the saved screws together, perhaps with a note indicating what they are for. In the excitement of getting your new Gemini up and running, it's easy to lose the fixing screws. Plus, you will want to re-position these screws if you ever ship your printer.

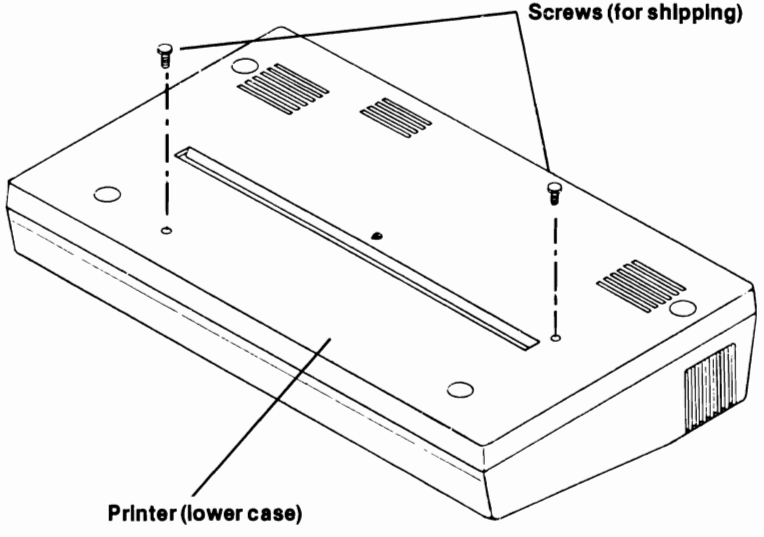



*Figure G-4. Turn your printer over and place it on a padded cushion, then remove the two "fixing" screws with a Phillips-head screwdriver.*

## ATTACHING THE PAPER SEPARATOR AND GUIDE

After you have removed the fixing screws, turn your printer back over so it's right-side-up.

In Chapter 2, you'll learn how to install the ribbon, set the print head gap, and load three types of paper. Those three types of paper are:

- ★ Individual sheets (such as stationery)
- ★ Sprocket-fed computer fanfold paper
- ★ Rolls of paper

For individual sheets of paper, you don't need any of these paper-feeding attachments. However, for sprocket-fed paper, you need to attach the paper separator and paper guide, as shown in Figures G-5 and G-6.

***Note:*** To insert these paper-feeding accessories, insert one side, then gently bend back the other protruding end until it slips into place.

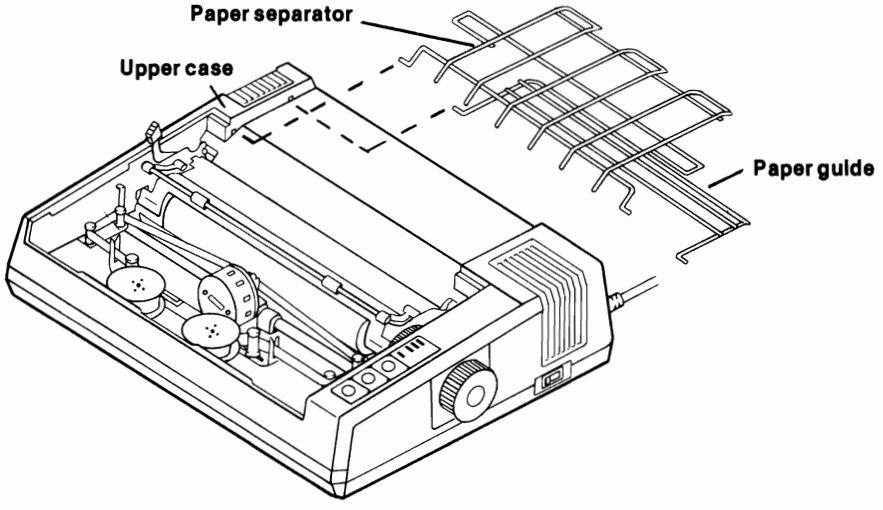



*Figure G-5. The paper guide and paper separator are inserted in the notches (right and left) indicated for one side in this illustration of the Gemini-10X*

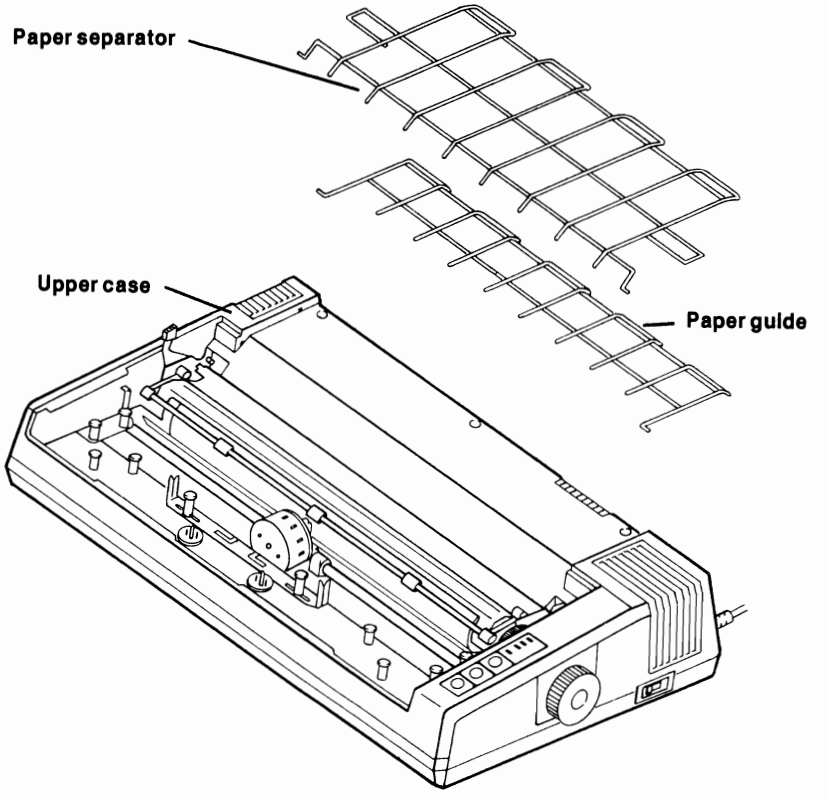



*Figure G-6. For the Gemini-15X the paper guide and separator are wider because the Gemini-15X is wider than the Gemini-10X. But these accessories attach in the same manner on both printers.*

***Another note:*** Be sure you don't get these racks in upside down. When you are finished, they should look like the Gemini's illustrated in Chapter 2 (Figures 2-4 through 2-6, and 2-8 through 2-13).

The purpose of these racks is to uniformly feed the paper (roll-type or sprocket-fed) between them and into the printing unit. Thus, you should keep these racks free from obstacles such as paper clips or pencils.

## ATTACHING THE ROLL PAPER HOLDER & SHAFT

Roll paper requires another accessory in addition to the paper separator and paper guide just introduced. As shown in Figure G-7 (for the Gemini-10X) the hooks of the holder should be inserted in the two holes on the undercarriage on the back side of the printer. (On the Gemini-15X the identical holder attaches in the same manner, but instead of in the middle, toward one side — away from the electrical plug.)

The paper holder is the rack that holds the roll of paper, much the same as some types of paper-towel dispensers. The roll of paper is placed on a holder shaft (as shown in Figure G-8).

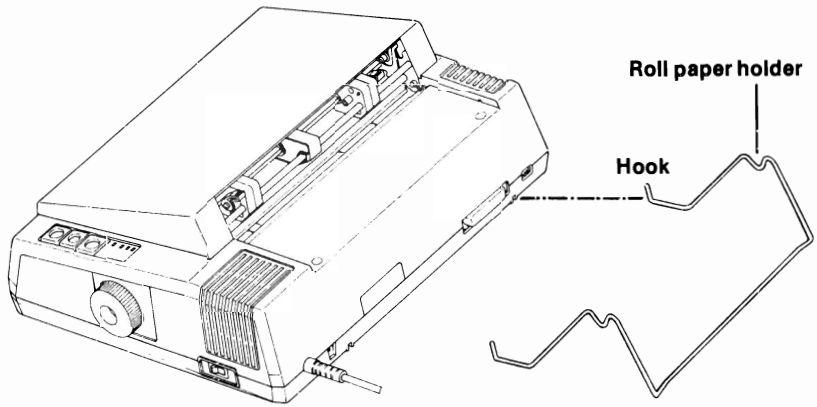



*Figure G-7. The roll-paper holder is attached to the back of the printer — in the center for the Gemini-10X and away from the electrical plug for the Gemini-15X.*

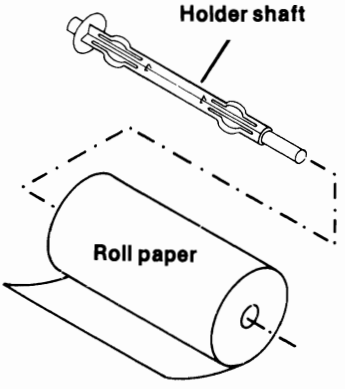



*Figure G-8. The holder shaft is inserted in the roll of paper, which is then mounted on the roll-paper holder, as shown in Figure G-9.*

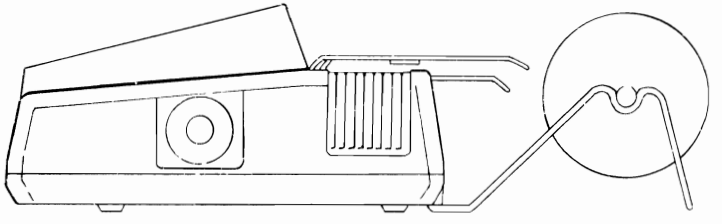
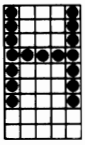


*Figure G-9. Side-view of the Gemini printer, showing the three paper-feed accessories in place.*

This concludes your introduction to the items contained in the Gemini printer package. Now you should turn to the beginning of this manual.

Chapter 1 will explain the basic parts of the printer. Chapter 2 will show you how to load the paper and the ink ribbon. Chapter 3 introduces the Gemini Self-Test. Chapter 4 shows you how to connect Gemini to your computer (supported by Appendices A through F, which are responsive to specific computer systems). Chapter 5 is a "quick course" in putting Gemini to work for you. The remaining chapters and appendices provide specialized and technical information. If you have a question, consult the index and the table of contents.

***Note:*** Before you go any further, have you reviewed and completed the **warranty registration**? It's a good warranty; you should take advantage of the opportunity and register.



# ASCII CODE CHARTS

## Standard ASCII Character Set

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NUL 0 | 16 | SP 32 | 0 48 | @ 64 | P 80 | ` 96 | p 112 |
| 1 | DC1 17 | ! 33 | 1 49 | A 65 | Q 81 | a 97 | q 113 |
| 2 | DC2 18 | " 34 | 2 50 | B 66 | R 82 | b 98 | r 114 |
| 3 | DC3 19 | # 35 | 3 51 | C 67 | S 83 | c 99 | s 115 |
| 4 | DC4 20 | $ 36 | 4 52 | D 68 | T 84 | d 100 | t 116 |
| 5 | 21 | % 37 | 5 53 | E 69 | U 85 | e 101 | u 117 |
| 6 | 22 | & 38 | 6 54 | F 70 | V 86 | f 102 | v 118 |
| BEL 7 | 23 | ' 39 | 7 55 | G 71 | W 87 | g 103 | w 119 |
| BS 8 | 24 | ( 40 | 8 56 | H 72 | X 88 | h 104 | x 120 |
| HT 9 | 25 | ) 41 | 9 57 | I 73 | Y 89 | i 105 | y 121 |
| LF 10 | 26 | * 42 | : 58 | J 74 | Z 90 | j 106 | z 122 |
| VT 11 | ESC 27 | + 43 | ; 59 | K 75 | [ 91 | k 107 | { 123 |
| FF 12 | 28 | , 44 | < 60 | L 76 | \ 92 | l 108 | ¦ 124 |
| CR 13 | 29 | – 45 | = 61 | M 77 | ] 93 | m 109 | } 125 |
| SO 14 | RS 30 | . 46 | > 62 | N 78 | ^ 94 | n 110 | ~ 126 |
| SI 15 | 31 | / 47 | ? 63 | O 79 | _ 95 | o 111 | DEL 127 |

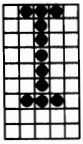
## Block Graphic and Special Character Set

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| 128 | 144 | ◜ 160 | $T_x$ 176 | Ā 192 | ¥ 208 | SP 224 | ⌜ 240 |
| 129 | DC1 145 | ◝ 161 | Å 177 | à 193 | Ä 209 | ■ 225 | ¯ 241 |
| 130 | DC2 146 | ◟ 162 | ø 178 | ç 194 | Ö 210 | ■ 226 | ⌝ 242 |
| 131 | DC3 147 | ◞ 163 | θ 179 | £ 195 | Ü 211 | ■ 227 | ⊤ 243 |
| 132 | DC4 148 | ↑ 164 | ℃ 180 | ā 196 | ¢ 212 | ■ 228 | ⊢ 244 |
| 133 | 149 | ↓ 165 | ℉ 181 | μ 197 | Ñ 213 | ▚ 229 | ⏐ 245 |
| 134 | 150 | ← 166 | Ω 182 | ° 198 | ä 214 | ▞ 230 | ⌞ 246 |
| BEL 135 | 151 | → 167 | ♂ 183 | ' 199 | ö 215 | ▬ 231 | ⌟ 247 |
| BS 136 | 152 | ○ 168 | Σ 184 | † 200 | ü 216 | ▬ 232 | ⊥ 248 |
| HT 137 | 153 | △ 169 | σ 185 | § 201 | β 217 | ▮ 233 | ⊣ 249 |
| LF 138 | 154 | ▽ 170 | ∞ 186 | Ē 202 | ē 218 | ▮ 234 | ┼ 250 |
| VT 139 | ESC 155 | ▷ 171 | π 187 | © 203 | é 219 | ▛ 235 | ◤ 251 |
| FF 140 | 156 | ◁ 172 | ± 188 | ¼ 204 | ú 220 | ▜ 236 | ◢ 252 |
| CR 141 | 157 | ◇ 173 | Ↄ 189 | x̄ 205 | è 221 | ▙ 237 | ◥ 253 |
| SO 142 | RS 158 | ◆ 174 | × 190 | ½ 206 | n̄ 222 | ▟ 238 | ◣ 254 |
| SI 143 | 159 | □ 175 | ÷ 191 | ‖ 207 | f 223 | ■ 239 | 255 |

## Italic ASCII Character Set

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| NUL 0 | 16 | SP 32 | *0* 48 | *@* 64 | *P* 80 | *`* 96 | *p* 112 |
| 1 | DC1 17 | *!* 33 | *1* 49 | *A* 65 | *Q* 81 | *a* 97 | *q* 113 |
| 2 | DC2 18 | *"* 34 | *2* 50 | *B* 66 | *R* 82 | *b* 98 | *r* 114 |
| 3 | DC3 19 | *#* 35 | *3* 51 | *C* 67 | *S* 83 | *c* 99 | *s* 115 |
| 4 | DC4 20 | *$* 36 | *4* 52 | *D* 68 | *T* 84 | *d* 100 | *t* 116 |
| 5 | 21 | *%* 37 | *5* 53 | *E* 69 | *U* 85 | *e* 101 | *u* 117 |
| 6 | 22 | *&* 38 | *6* 54 | *F* 70 | *V* 86 | *f* 102 | *v* 118 |
| BEL 7 | 23 | *'* 39 | *7* 55 | *G* 71 | *W* 87 | *g* 103 | *w* 119 |
| BS 8 | 24 | *(* 40 | *8* 56 | *H* 72 | *X* 88 | *h* 104 | *x* 120 |
| HT 9 | 25 | *)* 41 | *9* 57 | *I* 73 | *Y* 89 | *i* 105 | *y* 121 |
| LF 10 | 26 | * 42 | *:* 58 | *J* 74 | *Z* 90 | *j* 106 | *z* 122 |
| VT 11 | ESC 27 | *+* 43 | *;* 59 | *K* 75 | *[* 91 | *k* 107 | *{* 123 |
| FF 12 | 28 | *,* 44 | *<* 60 | *L* 76 | ** 92 | *l* 108 | *\|* 124 |
| CR 13 | 29 | *–* 45 | *=* 61 | *M* 77 | *]* 93 | *m* 109 | *}* 125 |
| SO 14 | RS 30 | *.* 46 | *>* 62 | *N* 78 | *^* 94 | *n* 110 | *~* 126 |
| SI 15 | 31 | */* 47 | *?* 63 | *O* 79 | *_* 95 | *o* 111 | DEL 127 |

# Notes

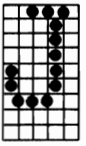


# ASCII CODE TABLES

| DEC | | 0 | 16 | 32 | 48 | 64 | 80 | 96 | 112 | 128 | 144 | 160 | 176 | 192 | 208 | 224 | 240 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | HEX | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | A | B | C | D | E | F |
| 0 | 0 | NUL | | SP | 0 | @ | P | ` | p | | | ⌐ | Tx | Ā | ¥ | SP | ┌ |
| 1 | 1 | | DC1 | ! | 1 | A | Q | a | q | | DC1 | ¬ | Å | à | Ä | ■ | ─ |
| 2 | 2 | | DC2 | " | 2 | B | R | b | r | | DC2 | ½ | ϕ | ç | Ö | ▪ | ┐ |
| 3 | 3 | | DC3 | # | 3 | C | S | c | s | | DC3 | ¼ | θ | £ | Ü | ▪ | ┬ |
| 4 | 4 | | DC4 | $ | 4 | D | T | d | t | | DC4 | ↑ | ℃ | ā | ¢ | ▪ | ├ |
| 5 | 5 | | | % | 5 | E | U | e | u | | | ↓ | °F | μ | Ñ | ▪▪ | │ |
| 6 | 6 | | | & | 6 | F | V | f | v | | | ← | Ω | ° | ä | ▪▪ | └ |
| 7 | 7 | BEL | | ' | 7 | G | W | g | w | BEL | | → | ð | ' | ö | ▬ | ┘ |
| 8 | 8 | BS | | ( | 8 | H | X | h | x | BS | | ○ | Σ | † | ü | ▬ | ┴ |
| 9 | 9 | HT | | ) | 9 | I | Y | i | y | HT | | △ | σ | § | β | ▮ | ┤ |
| 10 | A | LF | | * | : | J | Z | j | z | LF | | ▽ | ∞ | Ē | ē | ▮ | ┼ |
| 11 | B | VT | ESC | + | ; | K | [ | k | { | VT | ESC | ▷ | π | © | é | ▛ | ◤ |
| 12 | C | FF | | , | < | L | \ | l | ¦ | FF | | ◁ | ± | ¼ | ú | ▜ | ◢ |
| 13 | D | CR | | − | = | M | ] | m | } | CR | | ◇ | ↄ | x̄ | è | ▙ | ◥ |
| 14 | E | SO | RS | . | > | N | ^ | n | ~ | SO | RS | ◆ | × | ½ | n̄ | ▟ | ◣ |
| 15 | F | SI | | / | ? | O | — | o | DEL | SI | | □ | ÷ | ll | f | ■ | |

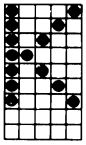
| 0 | 0 | NUL | | *SP* | *0* | *@* | *P* | *`* | *p* |
|---|---|---|---|---|---|---|---|---|---|
| 1 | 1 | | DC1 | *!* | *1* | *A* | *Q* | *a* | *q* |
| 2 | 2 | | DC2 | *"* | *2* | *B* | *R* | *b* | *r* |
| 3 | 3 | | DC3 | *#* | *3* | *C* | *S* | *c* | *s* |
| 4 | 4 | | DC4 | *$* | *4* | *D* | *T* | *d* | *t* |
| 5 | 5 | | | *%* | *5* | *E* | *U* | *e* | *u* |
| 6 | 6 | | | *&* | *6* | *F* | *V* | *f* | *v* |
| 7 | 7 | BEL | | *'* | *7* | *G* | *W* | *g* | *w* |
| 8 | 8 | BS | | *(* | *8* | *H* | *X* | *h* | *x* |
| 9 | 9 | HT | | *)* | *9* | *I* | *Y* | *i* | *y* |
| 10 | A | LF | | *** | *:* | *J* | *Z* | *j* | *z* |
| 11 | B | VT | ESC | *+* | *;* | *K* | *[* | *k* | *{* |
| 12 | C | FF | | *,* | *<* | *L* | ** | *l* | *¦* |
| 13 | D | CR | | *−* | *=* | *M* | *]* | *m* | *}* |
| 14 | E | SO | RS | *.* | *>* | *N* | *^* | *n* | *~* |
| 15 | F | SI | | */* | *?* | *O* | *_* | *o* | *DEL* |

**italic type**

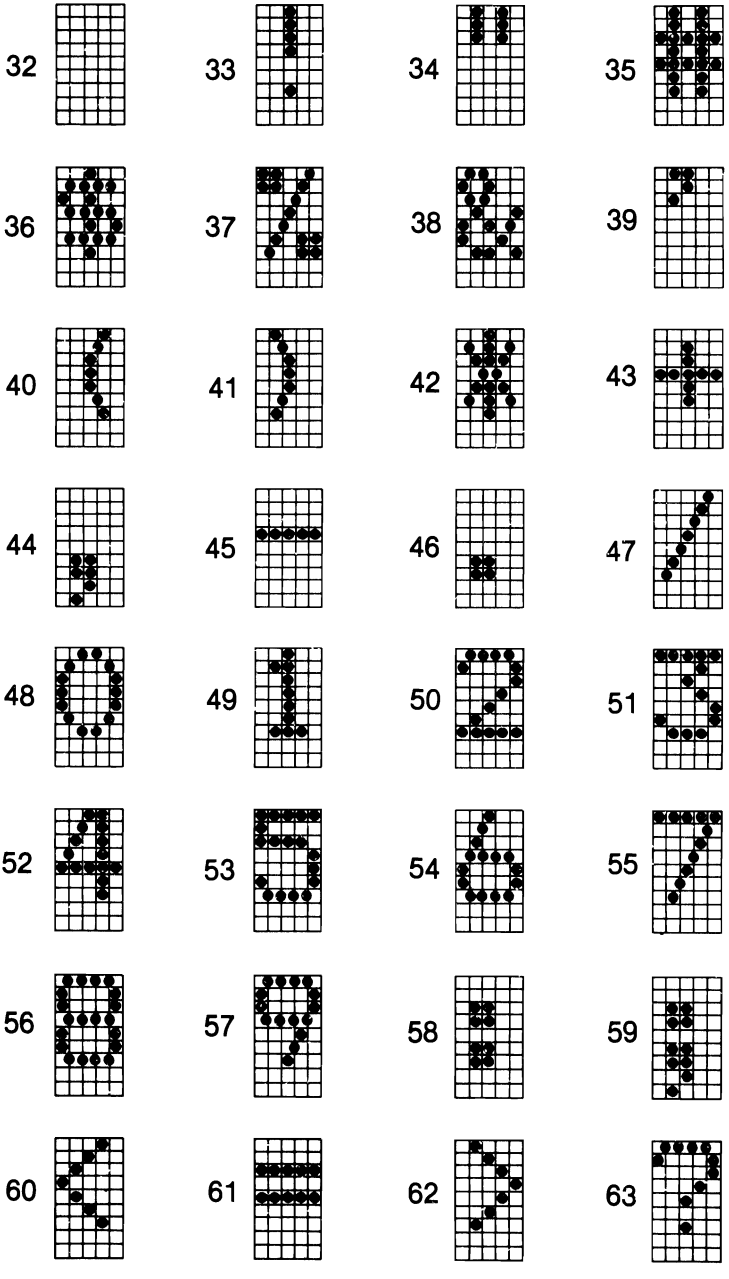


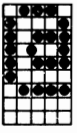
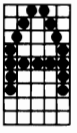
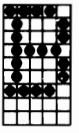
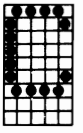
# Data Code List of International Character Set

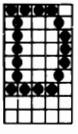
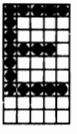
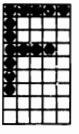
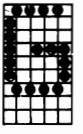
| DIP Switch No. | | | Control | HEX | 23 | 40 | 5B | 5C | 5D | 5E | 60 | 7B | 7C | 7D | 7E |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1-6 | 1-7 | 1-8 | Code | Country DEC | 35 | 64 | 91 | 92 | 93 | 94 | 96 | 123 | 124 | 125 | 126 |
| ON | ON | ON | ESC 7 0 | U.S.A. | # | @ | [ | \ | ] | ^ | ` | { | ¦ | } | ~ |
| OFF | ON | ON | ESC 7 1 | England | £ | @ | [ | \ | ] | ^ | ` | { | ¦ | } | ~ |
| ON | OFF | ON | ESC 7 2 | Germany | # | § | Ä | Ö | Ü | ^ | ` | ä | ö | ü | ß |
| OFF | OFF | ON | ESC 7 3 | Denmark | # | @ | Æ | Ø | Å | ^ | ` | æ | ø | å | ~ |
| ON | ON | OFF | ESC 7 4 | France | £ | à | ° | ç | § | ^ | ` | é | ù | è | ¨ |
| OFF | ON | OFF | ESC 7 5 | Sweden | # | É | Ä | Ö | Å | Ü | é | a | ö | å | ü |
| ON | OFF | OFF | ESC 7 6 | Italy | # | § | ° | ç | é | ^ | ù | à | ò | è | ì |
| OFF | OFF | OFF | ESC 7 7 | Spain | # | @ | ¡ | Ñ | ¿ | ^ | ` | ¨ | ñ | } | ~ |

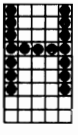
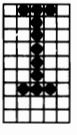
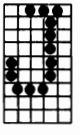
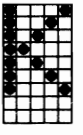


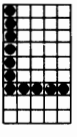
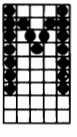
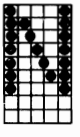
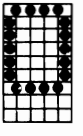
# CHARACTER STYLE TABLES

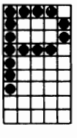
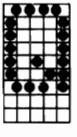
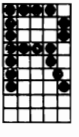
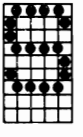


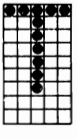
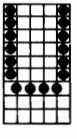
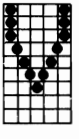
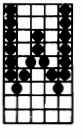
64 
65 
66 
67 

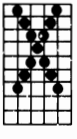
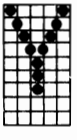
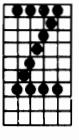
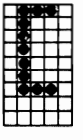
68 
69 
70 
71 

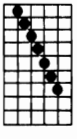
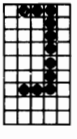
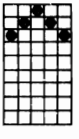
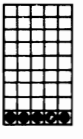
72 
73 
74 
75 

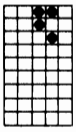
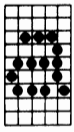
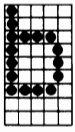
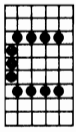
76 
77 
78 
79 

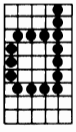
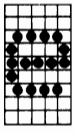
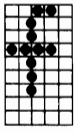
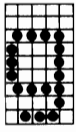
80 
81 
82 
83 

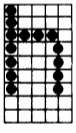
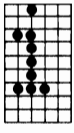
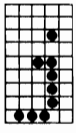
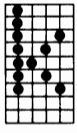
84 
85 
86 
87 

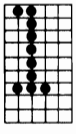
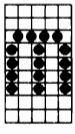
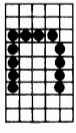
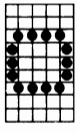
88 
89 
90 
91 

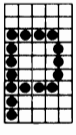
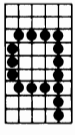
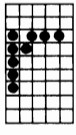
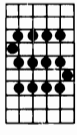
92 
93 
94 
95 

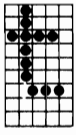
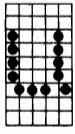
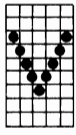
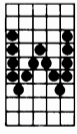
96 
97 
98 
99 

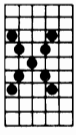
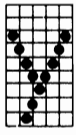
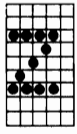
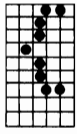
100 
101 
102 
103 

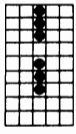
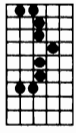
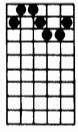
104 
105 
106 
107 

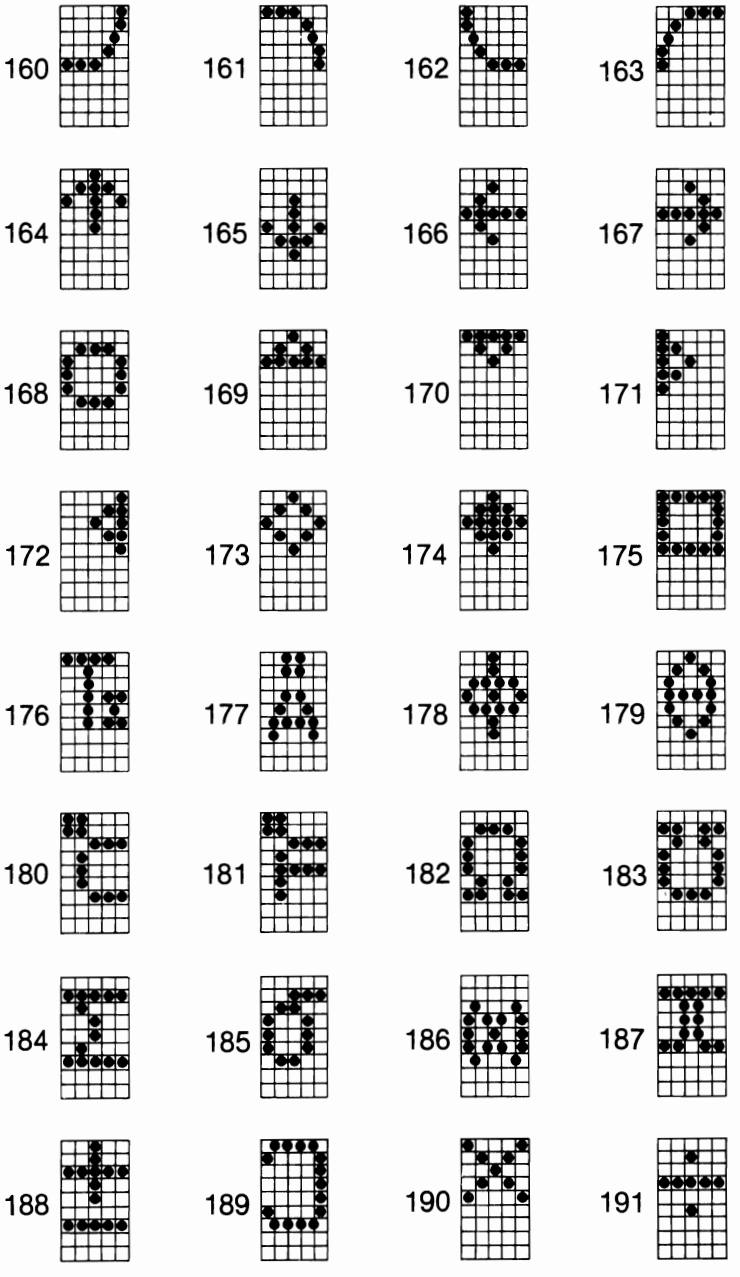
108 
109 
110 
111 

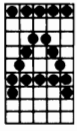
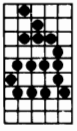
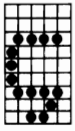
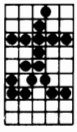
112 
113 
114 
115 

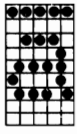
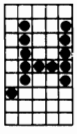
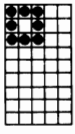
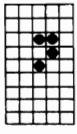
116 
117 
118 
119 

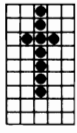
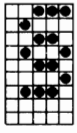
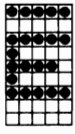
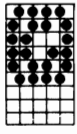
120 
121 
122 
123 

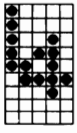
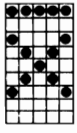
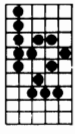
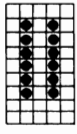
124 
125 
126 

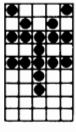
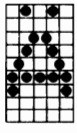
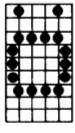
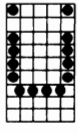

160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191

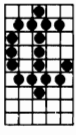
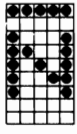
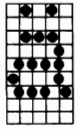
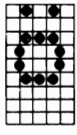
192 
193 
194 
195 

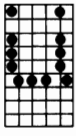
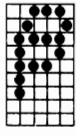
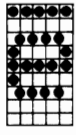
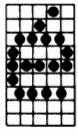
196 
197 
198 
199 

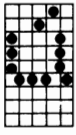
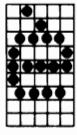
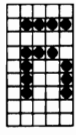
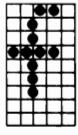
200 
201 
202 
203 

204 
205 
206 
207 

208 
209 
210 
211 

212 
213 
214 
215 

216 
217 
218 
219 

220 
221 
222 
223 

224

225

226

227

228

229

230

231

232

233

234

235

236

237

238

239

240

241

242

243

244

245

246

247

248

249

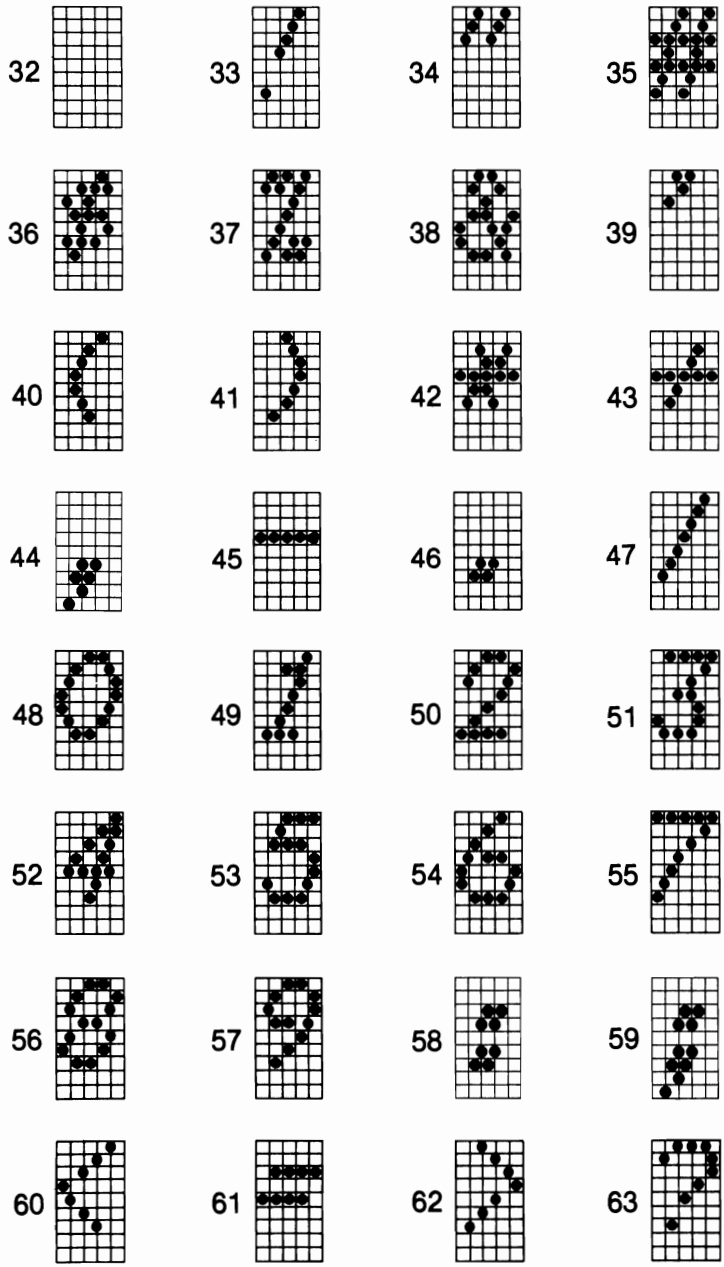
250

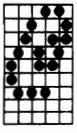
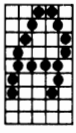
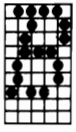
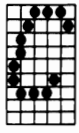
251

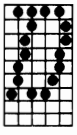
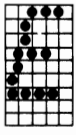
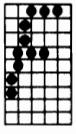
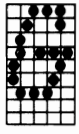
252

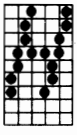
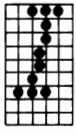
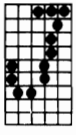
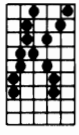
253

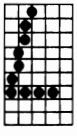
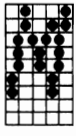
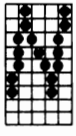
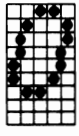
254

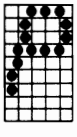
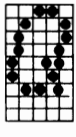
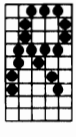
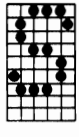

32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63

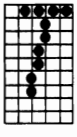
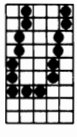
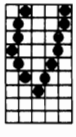
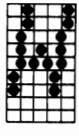
64 
65 
66 
67 

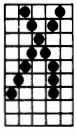
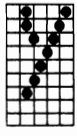
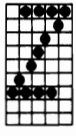
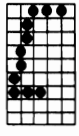
68 
69 
70 
71 

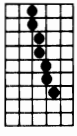
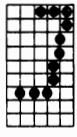
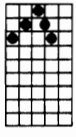
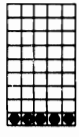
72 
73 
74 
75 

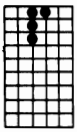
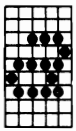
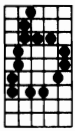
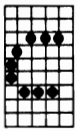
76 
77 
78 
79 

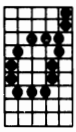
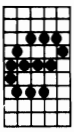
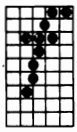
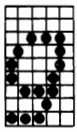
80 
81 
82 
83 

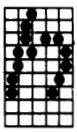
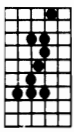
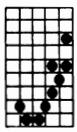
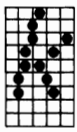
84 
85 
86 
87 

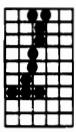
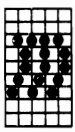
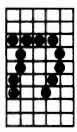
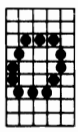
88 
89 
90 
91 

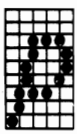
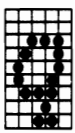
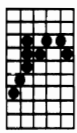
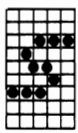
92 
93 
94 
95 

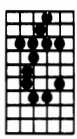
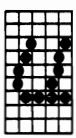
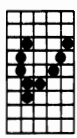
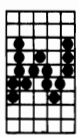
96 
97 
98 
99 

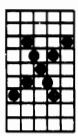
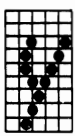
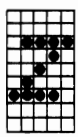
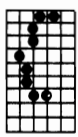
100 
101 
102 
103 

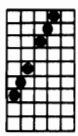
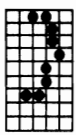
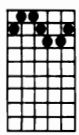
104 
105 
106 
107 

108 
109 
110 
111 

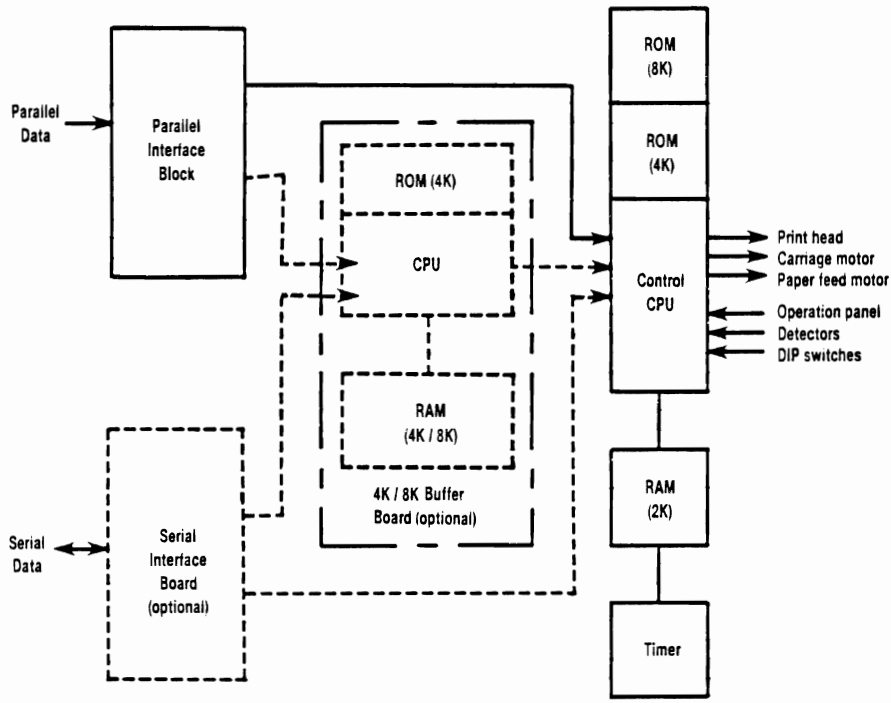
112 
113 
114 
115 

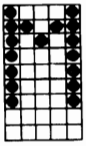
116 
117 
118 
119 

120 
121 
122 
123 

124 
125 
126 

# CONTROL CIRCUIT BLOCK DIAGRAM





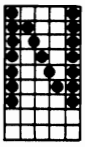
# FUNCTION CODE SUMMARY

| Control Code | Decimal | Hexi-decimal | Function |
|---|---|---|---|
| NUL | 0 | 00 | Used to terminate tab setting programming of ESC-D and ESC-P. |
| BEL | 7/135 | 07/87 | Buzzer sounds for ¼ second when this code is input. Input of "ESC Y 0" code negates BEL code. Input of "ESC Y1" makes BEL code effective. |
| BS | 8/136 | 08/88 | Buffer data is printed by this code and this code shifts the print start position to left side by one character space. |
| HT | 9/137 | 09/89 | Moves the print head to the next tab set position. |
| LF | 10/138 | 0A/8A | One line paper feed is done by an input of LF code after buffer data is printed. |
| VT | 11/139 | 0B/8B | Feed paper to the next VT position where programmed by "ESC P nn. . .n NUL." Default value is set every 6 lines. |
| FF | 12/140 | 0C/8C | Paper is fed to the print start line on next page by preset program (header line) after buffer data is printed. |
| CR | 13/141 | 0D/8D | CR code causes printout of buffer, and if DIP switch 2-4 (auto line feed) is in "on" position, CR and 1 line is fed automatically. |
| SO, ESC SO | 14/142 | 0E/8E | Turns on double-width character print mode. This mode is cancelled by using DC4 or CR. |
| SI, ESC S1 | 15/143 | 0F/8F | Turns on compressed mode (17 CPI). This mode cancelled by using DC2. |
| DC1 | 17/145 | 11/91 | This code causes printer to be selected. |
| DC2 | 18/146 | 12/92 | Cancels SI-mode. |
| DC3 | 19/147 | 13/93 | This code causes printer to be deselected. |
| DC4 | 20/148 | 14/94 | Cancels SO-mode. |
| RS | 30/158 | 1F/9E | This code ends the macro instruction. |
| DEL | 127 | 7F | Delete last character input to buffer. |

| Control Code | Decimal | Hexi-decimal | Function |
|---|---|---|---|
| ESC | 27/155 | 1B/9B | This code used for expansion control codes. |
| ESC # | 35 | 23 | Accepts 8th bit "as is" from host CPU. |
| ESC * 0 | 42 0 | 2A, 30 | Copies fonts in character ROM into download RAM |
| ESC * 1 n1 n2 m1 m2 | 42 1 n1 n2 m1 m2 | 2A, 31 n1 n2 m1 m2 | This code defines the download character into RAM |
| ESC $ 0 | 36 0 | 24,30 | Cancels the download character set. |
| ESC $ 1 | 36 1 | 24,31 | Selects the download character set |
| ESC-1 | 45 1 | 2D 01 | All data after this code is printed with underlining. |
| ESC-0 | 45 0 | 2D 00 | Cancels "ESC-1" underline mode. |
| ESC 0 | 48 | 30 | This code changes line feed length to 1/8". |
| ESC 1 | 49 | 31 | This code changes line feed length to 7/72". |
| ESC 2 | 50 | 32 | This code changes line feed length to 1/6". |
| ESC 3 n | 51 n | 33 n | This code changes line feed length to *n*/144". The value of *n* can be from 1 to 127. |
| ESC 4 | 52 | 34 | This code selects character ROM set of italic characters. |
| ESC 5 | 53 | 35 | This code cancels ESC 4. |
| ESC 7 n | 55 n | 37 n | This code selects the international character set. |
| ESC 8 | 56 | 38 | This code enables printing even though paper "out" sensor switch is "on." |
| ESC 9 | 57 | 39 | Cancels "ESC 8" mode. |
| ESC = | 61 | 3D | This code sets 8th bit to logic-0. |
| ESC > | 62 | 3E | This code sets 8th bit to logic-1. |
| ESC @ | 64 | 40 | Software "Reset" command, re-initialize all parameters to the power "on" condition. |
| ESC ! | 33 | 21 | This code selects the macro instruction. |
| ESC a n | 97 n | 61 n | This code sends a one-time-only vertical feed of n lines. |
| ESC b n | 98 n | 62 n | This code sends a one-time-only horizontal tab of n columns. |
| ESC A n | 65 n | 41 n | This code changes line feed length to *n*/72". Value of *n* can be from 1 to 127. |

| Control Code | Decimal | Hexi-decimal | Function |
|---|---|---|---|
| ESC B 1 | 66 1 | 42 01 | Sets 10 CPI. |
| ESC B 2 | 66 2 | 42 02 | Sets 12 CPI. |
| ESC B 3 | 66 3 | 42 03 | Sets 17 CPI. |
| ESC C n | 67 n | 43 n | Sets form length to *n* lines. Value of *n* can be from 1 to 127. |
| ESC C 0 n | 67 0 n | 43 30 n | Sets form length to *n* inches. Value of *n* can be from 1 to 32. |
| ESC D nn...n NUL | 68 n n 0 | 44 n n 00 | Clears current tab set and sets new tabs. Value of *n* is acceptable 1 to 255. (In case of 7-bit interface, 1 to 127.) |
| ESC E | 69 | 45 | Selects emphasized print mode. |
| ESC F | 70 | 46 | Cancels emphasized print mode. |
| ESC G | 71 | 47 | Selects double strike print mode. |
| ESC H | 72 | 48 | Cancels double strike print mode. |
| ESC J n | 74 n | 4A n | One-time line feed of *n*/144" |
| ESC K n1 n2 | 75 n1 n2 | 4B n1 n2 | 480 dots pitch of bit-image-column-scan mode is selected. Number of columns are determined *n*1 + *n*2 x 255. |
| ESC L n1 n2 | 76 n1 n2 | 4C n1 n2 | 960 dots pitch of bit-image-column-scan mode is selected. Number of columns are determined *n*1 + *n*2 x 255. |
| ESC M n | 77 n | 4D n | Sets LH-margin. Value of *n* can be up to maximum number of CPI. But in case of a 7-bit interface, up to 127. |
| ESC N n | 78 n | 4E n | Sets skip-over perforation to *n* lines. |
| ESC O | 79 | 4F | Resets skip-over perforation to zero lines. |
| ESC P nn...n NUL | 80 n n 0 | 50 n n 00 | Sets VT positions (vertical tab); default is every 6 lines. |
| ESC Q n | 81 n | 51 n | Sets RH-margin. Value of *n* can be up to maximum number of CPI. But in case of a 7-bit interface, up to 127. |
| ESC R n | 82 n | 52 n | Sets header line position. Value of *n* can be from 1 to 16. |
| ESC +...RS | 43, 30 | 2 B, 1 E | Defines the macro-instruction |
| ESC S 0 | 83 0 | 53 00 | Superscript mode is selected. |
| ESC S 1 | 83 1 | 53 01 | Subscript mode is selected. |
| ESC T | 84 | 54 | Cancels "ESC S" and "ESC U". |
| ESC U 1 | 85 1 | 55 01 | Selects uni-directional print mode. |
| ESC U 0 | 85 0 | 55 00 | Cancels uni-directional print mode. |

| Control Code | Decimal | Hexi-decimal | Function |
|---|---|---|---|
| ESC W 1 | 87 1 | 57 01 | Double-wide pitch print mode is selected. |
| ESC W 0 | 87 0 | 57 00 | Cancels "ESC W 1" mode. |
| ESC Y 1 | 89 1 | 59 01 | Enables buzzer. |
| ESC Y 0 | 89 0 | 59 00 | Disables buzzer. |
| ESC y n1 n2 m1 m2 | 121 n1 n2 m1 m2 | 79 n1 n2 m1 m2 | Prints dual-density graphics with double-speed. |
| ESC z n1 n2 m1 m2 | 122 n1 n2 m1 m2 | 7A n1 n2 m1 m2 | Prints quadruple-density graphics. |



# GEMINI'S PARALLEL INTERFACE

## Parallel Interface Specifications

The Gemini-10X and Gemini-15X printers have the following operational specifications:

**Parallel Interface Specs**

| | |
|---|---|
| Data transfer rate: | 1,000 to 6,000 CPS |
| Synchronization system: | Via externally supplied STROBE pulses |
| Handshake: | $\overline{\text{ACK}}$ and BUSY signals |
| Logic level: | Compatible with TTL level |

## The Connector Signals and Their Functional Descriptions

To understand the "mating" of Gemini with your computer, let's consider the function of the pins and the signals they convey. Figure N-1 provides a summary of the signals.

Pin 1 carries the $\overline{\text{STROBE}}$ pulse signal, which is normally kept at a high level (after the signal goes low, the data is read). Pins 2 through 9 carry the data, in 7-bit or 8-bit bytes. When the data is "1," the signal level is high, and the level goes down when the data is "0."

Pins 10 and 11 carry the "handshake" signals: $\overline{\text{ACK}}$ and BUSY. $\overline{\text{ACK}}$ is a pulse signal with a pulse-width of approximately 9 microseconds, which is generated when the data is received. When this pulse signal has been completed, the next data acceptance is permitted. BUSY is a signal at DC level that indicates the operating condition of the printer. When this signal goes low, the next data acceptance is permitted.

Pin 12 is the $\overline{\text{PAPER END}}$ signal connector. This DC signal is normally low, and goes high when there is a "no-paper" condition. (Note: DIP switch number 2-1 also controls the paper-empty status.)

Pin 13 connects another output DC signal, this one for indicating when the printer is online.

| Pin no. | Signal name | I/O | Function |
|---|---|---|---|
| 1 | $\overline{\text{STROBE}}$ | Input | Reads in the data |
| 2-9 | DATA 1-8 | Input | Communicates bytes of data |
| 10 | $\overline{\text{ACK}}$ | Output | Acknowledges receipt of data |
| 11 | BUSY | Output | Indicates receptiveness to more data |
| 12 | PAPER END | Output | Notifies that you're out of paper |
| 13 | SELECTED | Output | Online mode indicator |
| 14 | N/C | — | (unused) |
| 15 | N/C | — | (unused) |
| 16 | SIGNAL GND | — | Signal ground |
| 17 | CHASSIS GND | — | Frame ground |
| 18 | +5V DC | Output | External +5VDC Power (max. 50ma) |
| 19-30 | Twisted-pair returns for Pins 1-12 | — | Return signals, conveyed by twisted-pair cable |
| 31 | $\overline{\text{INPUT PRIME}}$ | Input | Resets printer: clears buffer and initializes |
| 32 | $\overline{\text{ERROR}}$ | Output | Indicates offline mode |
| 33 | EXT GND | — | External ground |
| 34-35 | N/C | — | (unused) |
| 36 | N/C | — | (TTL high level) |

*Figure N-1. The parallel interface signals and their names, pins, I/O classifications, and functional summary.*

The groundings are provided by three pins. Pin 16 is the signal ground, 17 is the chassis ground, and pin 33 is the external ground. Pin 18 carries 5 volts of direct current.

Twisted-pair return signals for pins 1 through 12 are conveyed by pins 19 through 30, respectively.

Pin 31, $\overline{\text{INPUT PRIME}}$, carries the signal for resetting the control logic. When this signal goes low, the printer is initialized and the memory buffer is cleared. Another output signal, $\overline{\text{ERROR}}$, is carried by pin 32. When Gemini is off-line or disabled by an abnormal state, the $\overline{\text{ERROR}}$ signal goes low.

Pins 14, 15, 34, 35, and 36 are unused.

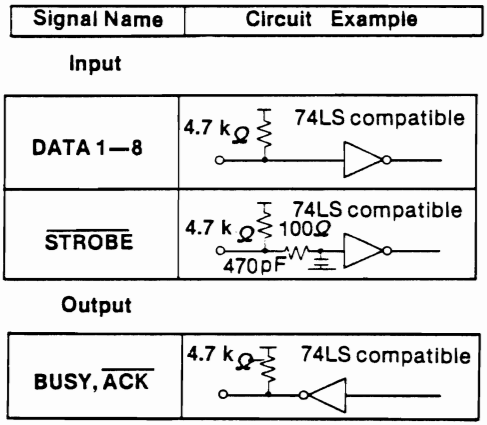



*Figure N-2. Sample circuit for parallel interface.*

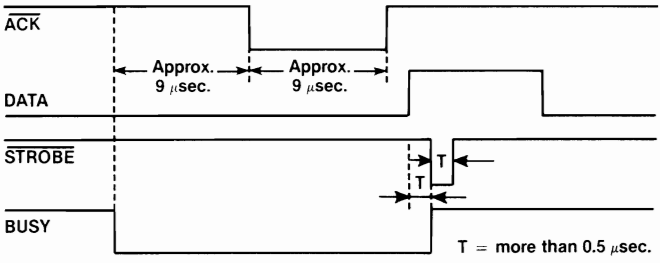
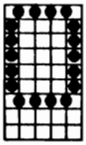



*Figure N-3. Data-transfer timing chart for parallel interface.*

# Notes



# Code Chart

This chart shows the code numbers in decimal, hexadecimal, and binary form.

| Decimal | Binary | Hexa-decimal | Decimal | Binary | Hexa-decimal |
|---|---|---|---|---|---|
| 0 | 00000000 | 00 | 36 | 00100100 | 24 |
| 1 | 00000001 | 01 | 37 | 00100101 | 25 |
| 2 | 00000010 | 02 | 38 | 00100110 | 26 |
| 3 | 00000011 | 03 | 39 | 00100111 | 27 |
| 4 | 00000100 | 04 | 40 | 00101000 | 28 |
| 5 | 00000101 | 05 | 41 | 00101001 | 29 |
| 6 | 00000110 | 06 | 42 | 00101010 | 2A |
| 7 | 00000111 | 07 | 43 | 00101011 | 2B |
| 8 | 00001000 | 08 | 44 | 00101100 | 2C |
| 9 | 00001001 | 09 | 45 | 00101101 | 2D |
| 10 | 00001010 | 0A | 46 | 00101110 | 2E |
| 11 | 00001011 | 0B | 47 | 00101111 | 2F |
| 12 | 00001100 | 0C | 48 | 00110000 | 30 |
| 13 | 00001101 | 0D | 49 | 00110001 | 31 |
| 14 | 00001110 | 0E | 50 | 00110010 | 32 |
| 15 | 00001111 | 0F | 51 | 00110011 | 33 |
| 16 | 00010000 | 10 | 52 | 00110100 | 34 |
| 17 | 00010001 | 11 | 53 | 00110101 | 35 |
| 18 | 00010010 | 12 | 54 | 00110110 | 36 |
| 19 | 00010011 | 13 | 55 | 00110111 | 37 |
| 20 | 00010100 | 14 | 56 | 00111000 | 38 |
| 21 | 00010101 | 15 | 57 | 00111001 | 39 |
| 22 | 00010110 | 16 | 58 | 00111010 | 3A |
| 23 | 00010111 | 17 | 59 | 00111011 | 3B |
| 24 | 00011000 | 18 | 60 | 00111100 | 3C |
| 25 | 00011001 | 19 | 61 | 00111101 | 3D |
| 26 | 00011010 | 1A | 62 | 00111110 | 3E |
| 27 | 00011011 | 1B | 63 | 00111111 | 3F |
| 28 | 00011100 | 1C | 64 | 01000000 | 40 |
| 29 | 00011101 | 1D | 65 | 01000001 | 41 |
| 30 | 00011110 | 1E | 66 | 01000010 | 42 |
| 31 | 00011111 | 1F | 67 | 01000011 | 43 |
| 32 | 00100000 | 20 | 68 | 01000100 | 44 |
| 33 | 00100001 | 21 | 69 | 01000101 | 45 |
| 34 | 00100010 | 22 | 70 | 01000110 | 46 |
| 35 | 00100011 | 23 | 71 | 01000111 | 47 |

| Decimal | Binary | Hexa-decimal | Decimal | Binary | Hexa-decimal |
|---|---|---|---|---|---|
| 72 | 01001000 | 48 | 116 | 01110100 | 74 |
| 73 | 01001001 | 49 | 117 | 01110101 | 75 |
| 74 | 01001010 | 4A | 118 | 01110110 | 76 |
| 75 | 01001011 | 4B | 119 | 01110111 | 77 |
| 76 | 01001100 | 4C | 120 | 01111000 | 78 |
| 77 | 01001101 | 4D | 121 | 01111001 | 79 |
| 78 | 01001110 | 4E | 122 | 01111010 | 7A |
| 79 | 01001111 | 4F | 123 | 01111011 | 7B |
| 80 | 01010000 | 50 | 124 | 01111100 | 7C |
| 81 | 01010001 | 51 | 125 | 01111101 | 7D |
| 82 | 01010010 | 52 | 126 | 01111110 | 7E |
| 83 | 01010011 | 53 | 127 | 01111111 | 7F |
| 84 | 01010100 | 54 | 128 | 10000000 | 80 |
| 85 | 01010101 | 55 | 129 | 10000001 | 81 |
| 86 | 01010110 | 56 | 130 | 10000010 | 82 |
| 87 | 01010111 | 57 | 131 | 10000011 | 83 |
| 88 | 01011000 | 58 | 132 | 10000100 | 84 |
| 89 | 01011001 | 59 | 133 | 10000101 | 85 |
| 90 | 01011010 | 5A | 134 | 10000110 | 86 |
| 91 | 01011011 | 5B | 135 | 10000111 | 87 |
| 92 | 01011100 | 5C | 136 | 10001000 | 88 |
| 93 | 01011101 | 5D | 137 | 10001001 | 89 |
| 94 | 01011110 | 5E | 138 | 10001010 | 8A |
| 95 | 01011111 | 5F | 139 | 10001011 | 8B |
| 96 | 01100000 | 60 | 140 | 10001100 | 8C |
| 97 | 01100001 | 61 | 141 | 10001101 | 8D |
| 98 | 01100010 | 62 | 142 | 10001110 | 8E |
| 99 | 01100011 | 63 | 143 | 10001111 | 8F |
| 100 | 01100100 | 64 | 144 | 10010000 | 90 |
| 101 | 01100101 | 65 | 145 | 10010001 | 91 |
| 102 | 01100110 | 66 | 146 | 10010010 | 92 |
| 103 | 01100111 | 67 | 147 | 10010011 | 93 |
| 104 | 01101000 | 68 | 148 | 10010100 | 94 |
| 105 | 01101001 | 69 | 149 | 10010101 | 95 |
| 106 | 01101010 | 6A | 150 | 10010110 | 96 |
| 107 | 01101011 | 6B | 151 | 10010111 | 97 |
| 108 | 01101100 | 6C | 152 | 10011000 | 98 |
| 109 | 01101101 | 6D | 153 | 10011001 | 99 |
| 110 | 01101110 | 6E | 154 | 10011010 | 9A |
| 111 | 01101111 | 6F | 155 | 10011011 | 9B |
| 112 | 01110000 | 70 | 156 | 10011100 | 9C |
| 113 | 01110001 | 71 | 157 | 10011101 | 9D |
| 114 | 01110010 | 72 | 158 | 10011110 | 9E |
| 115 | 01110011 | 73 | 159 | 10011111 | 9F |

| Decimal | Binary | Hexa-decimal | Decimal | Binary | Hexa-decimal |
|---|---|---|---|---|---|
| 160 | 10100000 | A0 | 184 | 10111000 | B8 |
| 161 | 10100001 | A1 | 185 | 10111001 | B9 |
| 162 | 10100010 | A2 | 186 | 10111010 | BA |
| 163 | 10100011 | A3 | 187 | 10111011 | BB |
| 164 | 10100100 | A4 | 188 | 10111100 | BC |
| 165 | 10100101 | A5 | 189 | 10111101 | BD |
| 166 | 10100110 | A6 | 190 | 10111110 | BE |
| 167 | 10100111 | A7 | 191 | 10111111 | BF |
| 168 | 10101000 | A8 | 192 | 11000000 | C0 |
| 169 | 10101001 | A9 | 193 | 11000001 | C1 |
| 170 | 10101010 | AA | 194 | 11000010 | C2 |
| 171 | 10101011 | AB | 195 | 11000011 | C3 |
| 172 | 10101100 | AC | 196 | 11000100 | C4 |
| 173 | 10101101 | AD | 197 | 11000101 | C5 |
| 174 | 10101110 | AE | 198 | 11000110 | C6 |
| 175 | 10101111 | AF | 199 | 11000111 | C7 |
| 176 | 10110000 | B0 | 200 | 11001000 | C8 |
| 177 | 10110001 | B1 | 201 | 11001001 | C9 |
| 178 | 10110010 | B2 | 202 | 11001010 | CA |
| 179 | 10110011 | B3 | 203 | 11001011 | CB |
| 180 | 10110100 | B4 | 204 | 11001100 | CC |
| 181 | 10110101 | B5 | 205 | 11001101 | CD |
| 182 | 10110110 | B6 | 206 | 11001110 | CE |
| 183 | 10110111 | B7 | 207 | 11001111 | CF |

| Decimal | Binary | Hexa-decimal | Decimal | Binary | Hexa-decimal |
|---|---|---|---|---|---|
| 208 | 11010000 | D0 | 232 | 11101000 | E8 |
| 209 | 11010001 | D1 | 233 | 11101001 | E9 |
| 210 | 11010010 | D2 | 234 | 11101010 | EA |
| 211 | 11010011 | D3 | 235 | 11101011 | EB |
| 212 | 11010100 | D4 | 236 | 11101100 | EC |
| 213 | 11010101 | D5 | 237 | 11101101 | ED |
| 214 | 11010110 | D6 | 238 | 11101110 | EE |
| 215 | 11010111 | D7 | 239 | 11101111 | EF |
| 216 | 11011000 | D8 | 240 | 11110000 | F0 |
| 217 | 11011001 | D9 | 241 | 11110001 | F1 |
| 218 | 11011010 | DA | 242 | 11110010 | F2 |
| 219 | 11011011 | DB | 243 | 11110011 | F3 |
| 220 | 11011100 | DC | 244 | 11110100 | F4 |
| 221 | 11011101 | DD | 245 | 11110101 | F5 |
| 222 | 11011110 | DE | 246 | 11110110 | F6 |
| 223 | 11011111 | DF | 247 | 11110111 | F7 |
| 224 | 11100000 | E0 | 248 | 11111000 | F8 |
| 225 | 11100001 | E1 | 249 | 11111001 | F9 |
| 226 | 11100010 | E2 | 250 | 11111010 | FA |
| 227 | 11100011 | E3 | 251 | 11111011 | FB |
| 228 | 11100100 | E4 | 252 | 11111100 | FC |
| 229 | 11100101 | E5 | 253 | 11111101 | FD |
| 230 | 11100110 | E6 | 254 | 11111110 | FE |
| 231 | 11100111 | E7 | 255 | 11111111 | FF |

# GLOSSARY

***Alternating current (AC)*** is the type of electricity used in household wiring. It's called "alternating" because the electrical flow rapidly alternates in direction. The Gemini printers operate using standard 120 volt AC, ± 10%.

***Alphanumeric*** is a character that is a letter of the *alpha*bet, a *numerical* symbol (a numeral), or any other symbol in a set of characters.

***American Standard Code for Informational Interchange (ASCII)*** is a widely accepted system for using the numerals 0 through 127 to represent alphanumeric and symbolic characters.

***Beginner's All-purpose Symbolic Instruction Code (BASIC)*** is a computer language found on personal computers and used (with some variations) for most small-computer programs.

***Binary*** is a term that, in a general sense, pertains to a choice or condition in which there are two possibilities. Specifically, it is used relative to computers to refer to the binary mathematical numeration system involving only two digits: one and zero. Computers code data in binary.

***Bit*** is the building block of computer memory that represents one of two values: zero or one, yes or no, on or off. A bit is a binary digit (0 or 1) that is grouped into a 7-unit or 8-unit entity called a byte, which conveys data in a computer system.

***Byte*** is the standard unit for measuring computer memory that consists of a grouping of adjacent binary digits (such as 00000111 to represent the quantity "7"), which are manipulated as a unit by the computer. A byte can represent one character.

***Buffer*** is a temporary storage area that is used to balance the difference between the operating speed of your computer and Gemini. The standard buffer capacity of Gemini is 816 bytes, with an optional capability of over 8,000 more bytes.

***Capacitor*** is an electrical device that usually consists of layers of conducting and nonconducting materials, used to store energy and to regulate electrical charges.

***Cathode ray tube (CRT)*** is the display screen on your microcomputer.

***Central processing unit (CPU)*** is the information-processing "traffic cop" of the printer. The ***CPU*** receives data from the interface circuit (from your computer) and transfers that data to the print-head circuitry.

***Character*** is the word for a representative symbol (such as the letter "A" or the numeral "5" or the "$" sign), which conveys all or part of an idea. The Gemini 10x printers offer 96 standard ASCII characters, 96 italic characters, 64 special characters, 32 block graphic characters, 96 downloadable characters as well as 88 foreign characters.

***Characters per second (CPS)*** is a measurement of printing speed, like miles per hour for an automobile. The Gemini printers have a print speed of 120 CPS at 10 ***characters per inch (CPI).***

***Chip*** is a small integrated circuit that is made from multiple layers of silicon and other materials, which conducts electricity in a manner that renders the device capable of handling computer-logic tasks.

***Computer*** is a device that can automatically process and execute a sequence of mathematical or logical operations.

***Computer language*** is the set of software instructions understood by the computer that allows the user to communicate human ideas through electronic devices.

***Data*** is a collection of factual information, converted to a form (coding) that allows it to be stored, transmitted, and manipulated by computers and their peripherals.

***Diode*** is a semiconductor, such as the ***light-emitting diode (LED)*** that is each lamp on the printer's control panel.

***Direct current (DC)*** is electrical current that flows in one direction ("directly"), such as with devices powered by flashlight batteries. The Gemini power unit converts the input AC power into voltages of ***+5V*** DC (which is used as a power supply for the CPU, ROM, and RAM), ***+8V*** DC (used to hold the positions of the motors and as a power supply to reset the timing circuit), ***+24V*** DC (used to drive the solenoids of the print head, and the motors).

***Down-Loading*** (Down-Loadable Character Sets) is the ability to transfer programmed character codes from a computer to a receptive peripheral device (Gemini-10X/15X) that contains storage RAM for such "down-loading" of specially created symbols.

***Dual In-line Package (DIP) switch*** is a set of "microswitches" that allows you to quickly convert your machine to perform important functions.

***Dump*** is a term for copying a storage file from the computer onto a line printer or other peripheral device.

***Escape character*** is the signal sent from your computer to Gemini, via your pressing the ***ESC*** key on the keyboard, that initiates one of the ***escape-function codes*** that control some of the printer's key operating parameters. In BASIC, sending ***CHR$(27)***, which is the code for ESC, initiates one of the ESC function control commands.

***File*** is a collection of related records that are treated as a single set of data for purposes of storage, retrieval, and transmission.

***Form Feed (FF)*** is a paper-advancing function that is initiated by pressing the FF button on the control panel while the printer is in the "local" mode. FF advances the paper around the platen and stops at the first line of the next page. FF may also be activated through software control.

***Fuse*** is an electrical safety device consisting of a metal wire (within a glass tube) that melts and interrupts the circuit when the amperage exceeds a predetermined point.

***Hardware*** is the set of tangible parts of a computer system: the computer, printer, and other peripherals — everything in a computer system except the information contained in that system.

***Input/Output (I/O)*** is the exchange of data between the computer and the printer. For example, your Gemini takes *input* from the computer in the form of data; when it cannot accept any more for the time being, Gemini sends *output* in the form of a "BUSY" signal, telling the computer to wait.

***Interface*** is the common boundary (physical connection) between two hardware items (such as your computer and Gemini) or between two systems. The interface permits the transfer of data and intelligent communication.

***Hexadecimal*** is the mathematical numeration system with a base of 16 (compared with 2 in the binary system and 10 in the decimal system). Appendix I presents the hexadecimal and decimal equivalents for the ASCII coded characters.

***K*** stands for ***kilobyte,*** specifically 1,024 bytes. Computer memory is often expressed in terms of K.

***LED,*** see Diode.

***Line Feed (LF)*** is a paper-advancing function that is initiated by pressing the LF button on the control panel of your Gemini. When this button is depressed, the paper advances one line; when this switch is held down, consecutive paper-feeds result for as long as the button is depressed.

***Local*** defines a condition in which the communication link between the printer and the computer has been interrupted and the printer is not on-line. The ***on-line control panel button*** is used to switch the printer between being local and on-line with the computer.

***Macro Instruction*** is the ability of a peripheral device (Gemini-10X/15X) to accept a single control code to do the work of a multi-command series of codes.

***Main memory*** refers to the electronic circuits that are instantaneously available to the computer, circuits containing storage of the software and data currently being used.

***Microprocessors*** are the "brains" of the personal computer. The most commonly used types, each about the size of a stack of three dimes, are the 8080, Z-80, and the 6502.

***Megabyte*** is equivalent to 1,024 kilobytes of memory. *Note:* Because computers work with binary numbers, K equals $2^{10}$ (or 1,024) bytes, rather than a thousand bytes as "K" would imply; similarly, a megabyte equals $2^{10}$ K.

***Off-line*** is a condition in which your printer can relate to itself but is unable to communicate with your computer.

***On-line*** is a condition in which the printer is interfaced with the computer and is ready to receive data transmission.

***Parallel transmission*** is a type of interface in which each byte of data is transmitted from the computer to the printer, arranged so that each bit is sent on a separate wire, side-by-side (in "parallel").

***Peripheral*** is any device used by a computer to communicate, including printers, monitors, and disk drives.

***Random Access Memory (RAM)*** is an array of semiconductor switches that stores information in the form of bits. RAM storage changes as the printer performs its tasks. The RAM is the main memory that the user can alter and might be called the "Read/Write Memory." In the Gemini printers, the RAM consists of 816 bytes of storage for data transferred from the interface circuit to the print head circuit.

***Read*** is to obtain data from a computer storage media or from user input.

***Read-Only Memory (ROM)*** is a type of main memory that comes from the manufacturer with data or programs that cannot be modified by the user. Unlike most RAMs, the contents of a ROM are not lost by turning the power "off."

***Resolution*** is the degree of detail that the user can obtain with different printing modes. The Gemini printers offer 60 x 72 low-resolution, 120 x 144 high-resolution and 240 x 144 ultra high resolution bit image graphics.

***Serial transmission*** is a type of communication in which each byte of data is transmitted from the computer to the printer, arranged so that each bit is sent on one wire, end-to-end ("serially").

***Software*** is the set of tasks or programs that makes hardware perform particular functions.

***Subscripts*** are those characters that are printed beneath the baseline for standard-size characters. The "2" is a subscript in $H_20$, the chemical notation for water.

***Superscripts*** are characters above the baseline of characters. The "2" is a superscript in $E = mc^2$, Einstein's simple equation for the relationship of energy to mass.

***Throughput*** is the total printing time, including data acquisition and line feed time.

***Write*** is to copy, generally from internal to external storage.

# *Index*


alphanumeric 71-72
Apple computers 151-164
ASCII code 58, 82, 226, 230
Atari computers 165-178

"back-pass" loading 28-30
backspacing 135
BASIC 59-60
bidirectional printing 128
binary 59, 93
bit image column scan
(*see* dot graphics)
bit-selection
(*see DIP* switches)
block graphics 84, 104-108
"bottom-pass" 28-31
buffer 7-8
buzzer control 139-140

casing removal 143-145
carriage return (CR) 68, 133
character strings, CHR$ 59-60
character styles 232-240
checklist for a new Gemini 217
circuit for parallel interface 248
clamp lever 23, 26
cleaning Gemini 143
column settings 84-85, 91-92
Commodore computers 179-191
condensed type 62-63, 66, 125
connecting cables 53-54
consumer response 261
control board 6-8
control codes 123-140, 242-245
back cover
control curcuit block
diagram 251
control panel 1-2, 5-6
cover 217-219
CPU 47-48

decimal codes 59, 226-230, 250-253
DIP switches 1-2, 49-53
dot graphics 90-97, 108, 136
double-strike mode 64, 67, 73, 85
downloadable characters 115-121,
137-138

elite printing (12 CPI)
65, 67, 85, 125
emphasized mode 66-67, 126-127
enlarged mode (5, 6, & 8.5 CPI)
66-67, 125
escape codes 123-140
exploded view 6-7

fixing screws 231
font style control 123-124
font pitch control 124-126
form feed (F.F.) 5-6, 130-132
French characters 99-100
function codes 58, 123-140
fuse replacement 144-145

German characters 100
glossary 254-257
grounding 3, 248

hexadecimal 59
horizontal print-head
position control 133-135

IBM-PC 192-193
italic characters 45, 66, 124
impact printing 4-5
initialization 55
interfacing 2, 47-48, 54
international character sets 51, 124, 231

line feed (L.F.) 5-6, 71, 129-130
line spacing 70-71
logic-seeking 6

macro commands 120-121, 139
maintenance 143-149
margin codes 134
mathematical and scientific characters 101-102
matrix printing 45, 72, 83-85
monetary characters 101

normal printing (10 CPI) 64-65, 67

online button 5-6, 62
on/off power switch 1-2
operating environment 9
Osborne computers 194-205

paper-out indicator 5
paper separator & guide 4, 217-219
paper loading 13-31
parallel interface 2, 48, 53-54, 246-249
pin-out charts 163, 177, 193, 195, 207
pins 246-248
platen knob 1-2
plotting characters 102-104
power unit 6-7
print commands (*see* BASIC and function codes)
print head 3-4, 37, 81-83, 146-147
programs in BASIC 63, 76-77, 89, 97, 105, 107, 111-112, 117-119, 121

RAM 241
ready lamp 5
release lever 16-17
reset function (*see* initialization)
ribbon loading 32-37
roll paper holder & shaft 4, 217-219
ROM 241

self-test 39-45
serial interface 2, 55
snap levers 14-15
Spanish characters 100
specifications for Gemini 10-11
STROBE pulse 246-249
subscripts 66, 73, 85, 128
superscripts 66, 73, 85, 127

timing chart 258
tractor unit 4-5, 14-16, 22-23
TRS-80 206-216
type styles 64-65, 71-74

underlining 84, 127
uni-directional printing 128

vertical spacing 85-90
vertical tabs 132

warnings for print head 17, 146
word-processing software 74-78

## *Consumer Response*

The editors and publisher of this Gemini User's Manual are interested in receiving any comments or suggestions that you, the consumer, may have regarding this publication.

Please direct your response to:

The Publisher
***Gemini User's Manual***
c/o Star Micronics, Inc.
P.O. Box 1630
El Toro, California, U.S.A. 92630

## STAR MICRONICS, INC.
## REGIONAL TECHNICAL SERVICES OFFICES

☆ **STAR MICRONICS, INC.**
P.O. Box 1630
El Toro, CA 92630
(714) 768-4340

☆ **STAR MICRONICS, INC.**
P.O. Box 612186
Dallas/Ft. Worth Airport, TX 75261
(214) 456-0052

☆ **STAR MICRONICS, INC.**
888 Washington Street
Suite 311
Dedham, MA 02026
(617) 329-8560

☆ **STAR MICRONICS, INC.**
500 Park Blvd.
Suite 645
Itasca, ILL 60143
(312) 773-3717

## Federal Communications Commission
## Radio Frequency Interference Statement

"This equipment generates and uses radio frequency energy and if not installed and used properly, that is, in strict accordance with the manufacturer's instructions, may cause interference to radio and television reception. It has been type-tested and found to comply with the limits for a Class B computing device in accordance with the specifications in Subpart J of Part 15 of FCC Rules, which are designed to provide reasonable protection against such interference in a residential installation. However, there is no guarantee that interference will not occur in a particular installation. If this equipment does cause interference to radio or television reception, which can be determined by turning the equipment off and on, the user is encouraged to try to correct the interference by one or more of the following measures:

- Reorient the receiving antenna
- Relocate the computer with respect to the receiver
- Move the computer away from the receiver
- Plug the computer into a different outlet so that computer and receiver are on different branch circuits.

If necessary, the user should consult the dealer or an experienced radio/television technician for additional suggestions. The user may find the following booklet prepared by the Federal Communications Commission helpful: "How to Identify and Resolve Radio-TV Interference Problems." This booklet is available from the US Government Printing Office, Washington, D.C., 20402, Stock No. 004-000-00345-4."

# DIP SWITCH GUIDE

| Dip Switch 1 Pin No. | Function | When "ON" | When "OFF" | Factory Setting |
|---|---|---|---|---|
| 1-1 | Selection of Character Set | Contingent on pin no. 4 (See Figure 4-5) | | ON |
| 1-2 | Print Mode | Normal/Emphasized | | ON |
| 1-3 | Set of Print Pitch | 10 CPI/17 CPI | | ON |
| 1-4 | Selection of Character Set | Contingent on pin no. 1 (See Figure 4-5) | | ON |
| 1-5 | Line Feed Value | 1/8 inch / 1/8 inch | | ON |
| 1-6 | Selection of International | Combinations of | | ON |
| 1-7 | Character Set and | switch settings | | ON |
| 1-8 | Form Length | produce varied output (See Figure 4-6) | | ON |

| Dip Switch 2 Pin No. | Function | When "ON" | When "OFF" | Factory Setting |
|---|---|---|---|---|
| 2-1 | No-paper detection | Signals paper-out/No signal | | ON |
| 2-2 | Buffer full | Prints at full buffer | Prints at each CR code | OFF |
| 2-3 | Bit selection | 7-bit interface/8-bit interface | | OFF |
| 2-4 | Automatic Line Feed | Automatic feed/No feed | | OFF |

DIP SWITCH STATUS CHART

| | ON | OFF |
|---|---|---|
| DIP SWITCH 1: | | |
| Pin 1: | _____ | _____ |
| Pin 2: | _____ | _____ |
| Pin 3: | _____ | _____ |
| Pin 4: | _____ | _____ |
| Pin 5: | _____ | _____ |
| Pin 6: | _____ | _____ |
| Pin 7: | _____ | _____ |
| Pin 8: | _____ | _____ |
| DIP SWITCH 2: | | |
| Pin 1: | _____ | _____ |
| Pin 2: | _____ | _____ |
| Pin 3: | _____ | _____ |
| Pin 4: | _____ | _____ |

You may wish to use this chart to keep track of the DIP Switch settings of the Gemini. This may come in handy as a good reference if things are not going as planned.

## Warranty

STAR MICRONICS, INC. warrants this product to be free from defects in material and workmanship for a period of 180 days (90 days for the print head and ribbon).

This warranty covers any and all product failure due to defects in material and workmanship within 180 days (90 days for print head and ribbon) from the date of original purchase. Star Micronics, Inc. will repair or, at its option, replace those components that prove defective, provided the product is returned (properly packed and shipping prepaid) to the nearest authorized Star Micronics Service Center, or to the dealer from whom the product was purchased.

Application for warranty coverage will not be granted by Star Micronics, Inc. if the defect or malfunction was caused by abuse or misuse of the product, operation not within the manufacturer's recommended specifications, or as a result of modification made by anyone other than an authorized service representative of Star Micronics, Inc.

Requests for warranty service should include a dated proof of purchase, a written description of the problem, and any accessory item (such as the interface/option character set). With your Gemini printer, you received a New Product Limited Warranty card and printer registration card. Please take a moment to fill them out, and mail your registration to the warranty processing center for Star Micronics, Inc.

***National Service Headquarters***
Star Micronics, Inc.
P.O. Box 1630
El Toro, California 92630

# QUICK REFERENCE GUIDE

**FUNCTION CONTROL COMMANDS:** See Chapter 9 and Appendix M for details.

| | |
|---|---|
| **NUL** | ends tab settings |
| **BEL** | sounds buzzer |
| **BS** | prints buffer data and moves PH one space left |
| **HT** | print at next tab setting |
| **LF** | one line paper feed |
| **VT** | feeds paper to next vertical tab, defaults to every 6th line |
| **FF** | feeds paper to start line of next page |
| **CR** | prints out buffer (see DIP Switch No. 2-4) |
| **SO, ESC SO** | turns on double width print mode |
| **SI, ESC SI** | turns on compressed print mode |
| **DC1** | selects printer |
| **DC2** | cancels SI mode |
| **DC3** | deselects printer |
| **DC4** | cancels SO mode |
| **RS** | ends macro instruction |
| **DEL** | deletes last character sent to buffer |
| **ESC** | initiates expansion control codes |
| **ESC #** | accepts 8th bit "as is" from host CPU |
| **ESC * 0** | copies fonts in ROM into download RAM |
| **ESC * 1** $n_1\ n_2\ m_1\ m_2$ | defines download characters into RAM |
| **ESC $ 0** | cancels download character set |
| **ESC $ 1** | selects download character set |
| **ESC - 1** | underlines |
| **ESC - 0** | cancels underlining |
| **ESC 0** | LF to 1/8" |
| **ESC 1** | LF to 7/72" |
| **ESC 2** | LF to 1/6" |
| **ESC 3 n** | LF to *n*/144" (*n* = 1 to 127) |
| **ESC 4** | initiates italic character |
| **ESC 5** | cancels ESC 4 |
| **ESC 7 n** | selects international character set |
| **ESC 8** | prints even with paper "out" |
| **ESC 9** | cancels ESC 8 |
| **ESC =** | sets 8th bit to logic zero |
| **ESC >** | sets 8th bit to logic one |
| **ESC @** | resets all parameters to initialized power "on" |
| **ESC !** | selects macro instruction |
| **ESC a n** | one-time only vertical feed |
| **ESC b n** | one-time only horizontal tab |
| **ESC A n** | changes LF to *n*/72" (*n* = 1 to 127) |
| **ESC B 1** | sets 10 CPI |
| **ESC B 2** | sets 12 CPI |
| **ESC B 3** | sets 17 CPI |
| **ESC C n** | sets form length to *n* lines (*n* = 1 to 127) |
| **ESC C 0 n** | sets form length to *n* inches (*n* = 1 to 32) |
| **ESC D** **nn...n NUL** | clears current horizontal tabs and sets new tabs (*n* = 1 to 255 for 8 bits, *n* = 1 to 127 for 7 bits) |
| **ESC E** | selects emphasized print mode |
| **ESC F** | cancels ESC E |
| **ESC G** | selects double-strike |
| **ESC H** | cancels ESC G |
| **ESC J n** | one-time line feed of *n*/144" |
| **ESC K n1 n2** | sets 60 dots/inch BICS, columns = *n*1 + *n*2 x 255 |
| **ESC L n1 n2** | sets 120 dots/inch of BICS |
| **ESC M n** | sets LH margin, where *n* is acceptable maximum characters/line |
| **ESC N n** | sets skip-over perf. to *n* lines |
| **ESC O** | resets skip-over perf. to zero lines |
| **ESC P** **nn...n NUL** | clears vertical tabs and resets to *n* |
| **ESC Q n** | sets RH margin |
| **ESC R n** | sets PH line position from top of form (*n* = 1 to 16) |
| **ESC +...RS** | Defines macro instruction |
| **ESC S 0** | selects superscript |
| **ESC S 1** | selects subscript |
| **ESC T** | cancels ESC S and ESC U |
| **ESC U 1** | selects unidirectional print mode |
| **ESC U 0** | deselects unidirectional print mode |
| **ESC W 1** | selects constant double-width print mode |
| **ESC W 0** | cancels ESC W 1 mode |
| **ESC Y 1** | enables buzzer |
| **ESC Y 0** | disables buzzer |
| **ESC y** $n_1\ n_2\ m_1\ m_2$ | dual-density graphics at double speed |
| **ESC z** $n_1\ n_2\ m_1\ m_2$ | quadruple-density graphics |

**LEGEND:** *BICS* = bit image column scan; *char.* = character(s); *CPI* = characters per inch; *DC* = device control; *ESC* = escape control; *LF* = line feed; *LH* = left-hand; *perf.* = perforation; *PH* = print head; *RH* = right-hand.

# GEMINI-10X/15X USERS MANUAL

star micronics inc

THE POWER BEHIND THE PRINTED WORD.

PRINTED IN JAPAN 12/83